Parametric approaches for modelling local structure tensor fields with applications to texture analysis

Roxana Gabriela Rosu

université de BORDEAUX

THÈSE

présentée pour obtenir le grade de

DOCTEUR DE L'UNIVERSITÉ DE BORDEAUX

ÉCOLE DOCTORALE DES SCIENCES PHYSIQUES ET DE L'INGÉNIEUR
SPÉCIALITÉ : AUTOMATIQUE, PRODUCTIQUE, SIGNAL ET IMAGE, INGÉNIERIE COGNITIQUE

par

Roxana-Gabriela ROŞU

PARAMETRIC APPROACHES FOR MODELLING LOCAL STRUCTURE TENSOR FIELDS WITH APPLICATIONS TO TEXTURE ANALYSIS

Sous la direction de : M. Jean-Pierre DA COSTA
co-encadrant : M. Marc DONIAS
Soutenue le 06 juillet 2018

Composition de jury :

Rapporteurs :	Ronan FABLET	Professeur, IMT Atlantique
	Antoine ROUEFF	Maître de Conférences, École Centrale de Marseille
Président de jury :	Ch. FERNANDEZ-MALOIGNE	Professeur, Univ. de Poitiers
Examinateur :	Samia BOUKIR	Professeur, Bordeaux INP
Directeur :	Jean-Pierre DA COSTA	Maître de Conférences, Univ. de Bordeaux
Co-encadrant :	Marc DONIAS	Maître de Conférences, Bordeaux INP

Acknowledgments

The years spent for the preparation of this thesis, challenging and gratifying on many different levels, have been rich. I think this accomplishment is the result of a collective effort and I wish to express my gratitude to all those who helped me face the challenges on the professional and scientific level and overcome all turbulences on the personal level as well.

First, I would like to express my sincere thanks and gratitude to my two thesis advisers, Jean-Pierre Da Costa and Marc Donias, for all the guidance provided during the preparation of this thesis. Thank you for all the scientific insights, comments and constructive feedback that contributed to the advancement and successful completion of this work. I express my appreciation for your availability and support, especially during the last and more challenging part of the thesis as well as for the human qualities kindly manifested in addition to the scientific ones all along these years.

I would like to express my sincere gratitude to all the members of the thesis committee who made me the honour of evaluating my work. Thank you, Antoine Roueff and Ronan Fablet, for having accepted to read my thesis and for your rich comments and invaluable feedback that have opened interesting prospects for future work. I extend my gratitude to Christine Fernandez-Maloigne and to Samia Boukir for their positive evaluation and for all the useful remarks that helped improve the quality of my thesis report.

My gratitude goes to Monica Borda and Jean-François Giovannelli for establishing the collaboration between the Technical University of Cluj-Napoca and the University of Bordeaux, that allowed me to do a part of my master studies in Bordeaux.

A warm thought to all my colleagues from the Signal and Image Processing group that made my years in the IMS Laboratory more pleasant and rich in terms of multi-cultural experiences. A thought of gratitude to Yannick Berthoumieu for welcoming me in the group and for all the support offered during my teaching experience. A special thank you and many *plăcinte cu mere* to Lionel Bombrun for the kindness and availability showed in answering my questions. I extend my gratitude to Salem Said for the useful exchanges that helped me see the Riemannian geometry in a less scary way. Thank you for having taken the time to read a chapter of my thesis and for all the useful feedback. In addition, I wish to thank the intern-ship, Ph.D. and post-doc students for all the good moments spent together. A particular thank you to Florian and to Florent, the two agronomists in the team, for keeping my plants alive during my absence times from the lab. Thank you to Jessica, Ouiame and Mireille for all the good moments and laughs under the close supervision of Vlady. A warm thought to Sara and Yousra for your useful help during the preparation of the *pot de thèse*.

I am grateful (and Licornitto [1] as well) that some encounters transformed into invaluable friendships. Thank you, Barna, for the bike rides and for always finding activities and small trips ideas. Warm thanks go to Marco for being a true *amic voinic*, for your encouragements, for all the fun

[1] trying to define the indefinable: that creative and playful *je ne sais quoi* that approaches life with curiosity and a no fear of mistakes attitude in a never winning attempt of decrypting its mysteries, while creating some new ones in the process.

moments in Licornitto's spirit and for the shared taste for *n'importe quoi* moments. A warm thought goes to Mariem, for both the good and bad moments spent together during the last year: the good laughs and mms moments are equally appreciated. My gratitude goes to Ioana for your gentleness and kindness, it was a pleasure to share the same office and may I officially be excused for interrupting you 10 times a day with random conversation topics (they were all Licornitto's ideas). Andreia, I thank you for all our conversations and philosophical debates, and for your open-minded vision on people and things, for rather seeing something as extravagant than weird – just one small detail that makes a difference. I extend my gratitude to Léa, it was a pleasure co-supervising you during your intern-ship and later discovering that we share common interests and projects other than the structure tensor. Given our centenarian goals, I would say we have some time to implement all that. A warm grateful thought for Jean-Marie as well, for the rich scientific discussions, the completely objectively-based pride in your Breton heritage and for the vivid teachings of all sort of useful expressions in French, among which *ça ramone du mammouth* seems to be particularly useful. Now I also find myself asking every morning if I have *la patate* and if I am *au taquet*.

Many thanks to Lili, for encouraging me and for all the small enjoyable trips we have made together. My sincere gratitude goes to Alexandra and Radu, who almost adopted me at my arrival in Bordeaux and who certainly made my integration much easier and to lovely Anastasia, whom I have the honour to be the godmother to. I wish to express a heart-felt thank you to Camelia and Lucian Chirila, for all your support, kindness and generosity. A warm thought of gratitude for Aniela, for supporting me with patience and kindness in one of the most challenging but rewarding journeys. A sincere thank you to Christos, Iza and Janos, for all the good moments spent together.

I would like to address a huge thank you to my Romanian friends back home who proved that distance is no threat to true friendships. A sincere heart-felt thank you to Iulia for being the most wonderful supporter anyone can wish for. You probably encouraged me more than I did myself and believed in me throughout all my moments of doubt (and we both know there were plenty of them). My deep gratitude goes to Cristina, for your infinite support, for always finding some humour in the darkest of times (or some tissues when humour didn't work) and for constantly teaching me what friendships is all about. Your presence at my thesis defence was much appreciated (and all your involvement in the messy unexpected flow of events that followed – thanks for your friendliness in approaching my sometimes complete unrealistic ideas – I want to learn that technique, too). Friends like you two make me be and want to be a better person. Thank you, Casi, for always being there for me and for the natural and immediate way of continuing a conversation from where we left it some months ago.

Last but not least, I wish to express my profound heart-felt gratitude to my parents. Thank you for your love, for believing in me, for always being on my side and for accepting my decisions even when they do not correspond to your deepest desires.

List of Abbreviations

AI	Affine-Invariant
ARR	Average Retrieval Rate
ANMRR	Average Normalized Modified Retrieval Rate
BIC	Bayesian Information Criterion
DTI	Diffusion Tensor Imaging
DT-MRI	Diffusion Tensor - Magnetic Resonance Imaging
FPS	Fourier Power Spectrum
GCM	Multivariate Gamma Distribution on Gaussian Copula
GLCM	Gray Level Cooccurrence Matrix
HRTEM	High Resolution Transmission Electron Microscopy
KLD	Kullback - Leibler Divergence
k-NN	k - Nearest Neighbours
LCPDF	Local Conditional Probability Density Function
LE	Log-Euclidean
LF	Lattice Fringe
LBP	Linear Binary Patterns
LST	Local Structure Tensor
MCMC	Markov Chain Monte Carlo
MLE	Maximum Likelihood Estimation
MRF	Markov Random Field
OA	Overall Accuracy
PA	Producer's Accuracy
PyC	Pyrocarbon

RGB	Red Green Blue
RI	Rotation Invariant
RS	Remote Sensing
SA	Simulated Annealing
SAR	Synthetic Aperture Radar
SCM	Multivariate Gaussian Model with SCM (Sample Covariance Matrix) estimator for the Covariance Matrix
SIFT	Scale Invariant Feature Transform
SIRVg	Spherically Invariant Random Vector with a multivariate Gaussian model
SPD	Symmetric Positive Definite
SVM	Support Vector Machine
UA	User's Accuracy
VHR	Very High Resolution
VLAD-PQ	Product-Quantized binary version of the Vector of Locally Aggregated Descriptors

Approches Paramétriques pour la Modélisation de Champs de Tenseurs de Structure Locaux et Applications en Analyse de Texture

- résume étendu -

"Les racines des mots sont-elles carrées?"

—Eugène Ionesco, Dramaturge Roumano-Français

Contexte

Le tenseur de structure local (TSL) [Bigün 1987, Knutsson 1989] est un outil communément utilisé en vision par ordinateur pour décrire la structure locale d'une image. Il peut être utilisé pour différentes tâches telles que la détection des bords ou des coins d'objets présents sur une image [Harris 1988] ou le filtrage et le rehaussement d'images [Weickert 2014]. Le tenseur de structure est principalement utilisé pour décrire la géométrie d'images texturées, en particulier celles présentant une tendance directionnelle.

Défini en chaque pixel d'une image comme étant la matrice de covariance locale des gradients, le tenseur de structure permet d'extraire une direction dominante ainsi qu'un degré d'anisotropie indiquant la cohérence des gradients autour de la direction estimée. Si le tenseur de structure est un outil couramment employé en analyse [Rousson 2003, Pham 2015b] et parfois en synthèse de texture [Akl 2015], sa modélisation probabiliste a rarement été abordée dans la littérature. Dans un contexte d'estimation et de segmentation jointes, une approche reposant sur la modélisation du champ des orientations locales par des distributions de Von Mises [Mardia 2000] a été précédemment proposée [Da Costa 2012]. Dans un contexte de segmentation, une autre approche s'appuie conjointement sur la texture et la couleur [de Luis-García 2008]. Il s'agit d'utiliser la divergence de Kullback-Leibler entre deux modèles Gaussiens comme mesure de dissimilarité entre tenseurs, considérés comme étant chacun la matrice de covariance d'une loi normale.

En tant que matrices de covariance, les tenseurs de structure appartiennent à la variété rie-

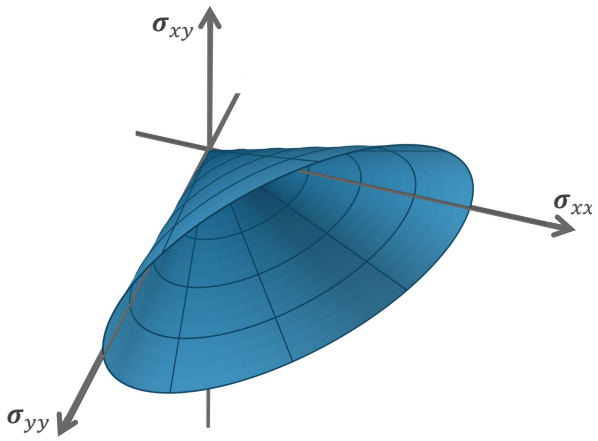


Figure 1: Sous-espace occupé par les matrices symmétriques semi-définies positives dans l'espace vectoriel $\mathbf{R}^3$ correspondant aux matrices symétriques de taille 2×2.

mannienne des matrices symétriques semi-définies positives. Comme conséquence, l'utilisation du TSL pose un problème formel de géométrie de l'information et ne peut se faire en s'appuyant sur les outils classiques de la géométrie euclidienne. Comme dans le cas de tenseurs de diffusion [Lenglet 2006, Dryden 2009], la manipulation de TSL doit s'appuyer sur des outils et des métriques non-euclidiens, adaptés à la géométrie de la variété Riemannienne. Le sous espace occupé par les matrices symétriques semi-définies positives dans l'espace de matrices symétriques de forme générale $\begin{pmatrix} \sigma_{xx} & \sigma_{xy} \\ \sigma_{xy} & \sigma_{yy} \end{pmatrix}$ est illustré dans la Figure 1.

Objectifs

Cette thèse porte sur des méthodes probabilistes pour la modélisation de champs de tenseurs de structure calculés sur des images texturées. Les outils euclidiens s'avérant inadaptés au calcul tensoriel, des alternatives non-euclidiennes sont étudiées. Plus précisément, des espaces métriques riemanniens sont explorés dans le but de construire des canevas méthodologiques adaptés à la description statistique de champs de TSL. Des approches pour décrire les lois marginales de tenseurs de structure sont analysées dans un premier temps. Ensuite, des approches de modélisation de distributions jointes de TSL sont explorées. Ces dernières permettent la description des dépendances spatiales et multi-échelles d'un champ de tenseurs de structure. De plus, des mesures de dissimilarité sont associées aux modèles statistiques proposés permettant de comparer les champs de tenseurs de structure calculés sur différentes images texturées.

Les potentialités des méthodes paramétriques proposées sont illustrées dans deux contextes applicatifs. Dans un premier temps, ces modèles sont employés dans un objectif de reconnaissance de textures, au travers de schémas d'indexation et de classification d'images. Puis, les capacités descriptives de ces modèles sont évaluées de façon plus approfondie au travers d'expérimentations

de synthèse de champs de TSL.

Méthodologie

Deux canevas méthodologiques riemanniens sont explorés pour la manipulation de matrices de covariance symétriques définies positives (SDP). Ils reposent respectivement sur les métriques affine-invariante (AI) et log-euclidienne (LE), disposant des propriétés d'invariance les plus complètes [Dryden 2009, Jayasumana 2013] parmi toutes les alternatives non-euclidiennes étudiées. Chaque cadre méthodologique s'articule autour de la définition d'un modèle gaussien. Afin d'enrichir les capacités descriptives de ces modèles, leurs lois de mélange associées sont également exploitées. Des solutions pour évaluer la dissimilarité entre les différents modèles estimés sont aussi proposées.

Le modèle affine-invariant de matrices de covariances

Soit $\mathcal{P}_2$ l'espace de matrices 2×2 réelles et symétriques définies positives. Un modèle Gaussien Riemannien, proposé par Said et al. [Said 2017], est défini sur l'espace $\mathcal{P}_2$. Sa densité de probabilité est :

$$p(\mathbf{Y}|\mathbf{M}, \sigma) = \frac{1}{Z(\sigma)} \exp\left[-\frac{d^2(\mathbf{Y}, \mathbf{M})}{2\sigma^2}\right], \tag{1}$$

où $\mathbf{M} \in \mathcal{P}_2$ et $\sigma \in \mathbb{R}_+^*$ sont le barycentre et la dispersion de la distribution et $Z(\sigma)$ est une constante de normalisation dépendant uniquement du paramètre de dispersion σ. La fonction $d : \mathcal{P}_2 \times \mathcal{P}_2 \to \mathbb{R}_+$ représente la distance géodésique appelée également distance de Rao [James 1973] :

$$d^2(Y_1, Y_2) = \mathrm{Tr}\left[\mathrm{logm}^2\left(Y_1^{-\frac{1}{2}} Y_2 Y_1^{-\frac{1}{2}}\right)\right], \tag{2}$$

où logm désigne le logarithme matriciel.

La formulation du maximum de vraisemblance ne permet pas d'aboutir à des expressions analytiques simples pour l'estimation des paramètres. En revanche, des algorithmes d'estimation récursifs sont utilisés. A cet effet, l'algorithme de *descente de gradient Riemannien* et l'algorithme de *Newton-Raphson* sont employés respectivement pour l'estimation du barycentre et de la dispersion du modèle gaussien riemannien.

Le modèle log-euclidien de matrices de covariances

L'espace log-euclidien [Arsigny 2005] autorise la représentation vectorielle d'une matrice de covariance et sa manipulation dans le domaine des logarithmes matriciels par les outils classiques de la géométrie euclidienne. En conservant une grande majorité des propriétés d'invariance associées à la métrique AI, la métrique LE présente l'avantage de mettre en œuvre des algorithmes de complexité calculatoire bien plus faible.

Le passage d'une matrice de covariance $\mathbf{Y}$ à l'espace log-euclidien repose sur le calcul de son logarithme matriciel. Le calcul du logarithme matriciel d'une matrice SDP aboutit toujours à une matrice symétrique dont les trois composantes peuvent être écrites sous la forme vectorielle $\vec{\mathbf{Y}}_{\mathrm{LE}}$:

$$\mathbf{Y}_{\mathrm{LE}} = \mathrm{logm}(\mathbf{Y}) = \begin{pmatrix} y_{\mathrm{LE}_{xx}} & y_{\mathrm{LE}_{xy}} \\ y_{\mathrm{LE}_{xy}} & y_{\mathrm{LE}_{yy}} \end{pmatrix}, \vec{\mathbf{Y}}_{\mathrm{LE}} = \left[y_{\mathrm{LE}_{xx}},\ \sqrt{2} y_{\mathrm{LE}_{xy}}, y_{\mathrm{LE}_{yy}} \right]^{\dagger}, \tag{3}$$

où † est l'opérateur transposé.

Un modèle gaussien multivarié de dimension 3 est considéré pour caractériser des échantillons de matrices de covariance de taille 2×2 dans l'espace log-euclidien. Ce modèle est paramétré par une moyenne $\boldsymbol{\mu}$ de taille 3×1 et une matrice de covariance $\boldsymbol{\Sigma}$ de taille 2×2.

L'estimation des paramètres des lois peut alors se faire au sens du maximum de vraisemblance. L'utilisation des formes analytiques des estimateurs disponibles pour les lois normales multivariées permet de réduire de façon significative la complexité calculatoire par rapport aux approches affine-invariantes évoquées.

Pour les deux modèles gaussiennes associées aux espaces AI et LE, des modèles de mélanges sont également considérées. Dans ces cas, l'estimation de paramètres repose sur l'algorithme *Expectation Maximisation*.

La version symétrique de la divergence de Kullback-Leibler est considérée pour évaluer la dissimilarité entre deux modèles statistiques définis sur l'espace AI ou sur l'espace LE. Le modèle Gaussien multivarié est le seul parmi ceux considérés pour lequel la divergence de Kullback Liebler est disponible sous une forme analytique. Dans toutes les autres cas, des solutions d'estimation basées sur un échantillonnage de Monte Carlo sont par conséquent proposées pour évaluer les divergences.

Utilisation de modèles pour la description de champs de TSL des images texturées

Les approches méthodologiques proposées sont utilisées pour la modélisation de champs de tenseurs de structure calculés sur des textures réelles.

Le calcul de tenseur pour une image I repose tout d'abord sur l'estimation de son champ gradient $\boldsymbol{\nabla} I$. Dans ce mémoire le champ gradient est obtenu en convoluant l'image avec des filtres dérivatifs gaussiens G_x et G_y, d'écart type σ_G :

$$\boldsymbol{\nabla} I = \left[I_x, I_y \right]^{\dagger} = \left[I * G_x, I * G_y \right]^{\dagger}, \tag{4}$$

où $*$ désigne l'opération de convolution et I_x et I_y sont respectivement les dérivées partielles horizontales et verticales de l'image $\mathbf{I}$. Ensuite, le tenseur de structure est estimé en chaque point en

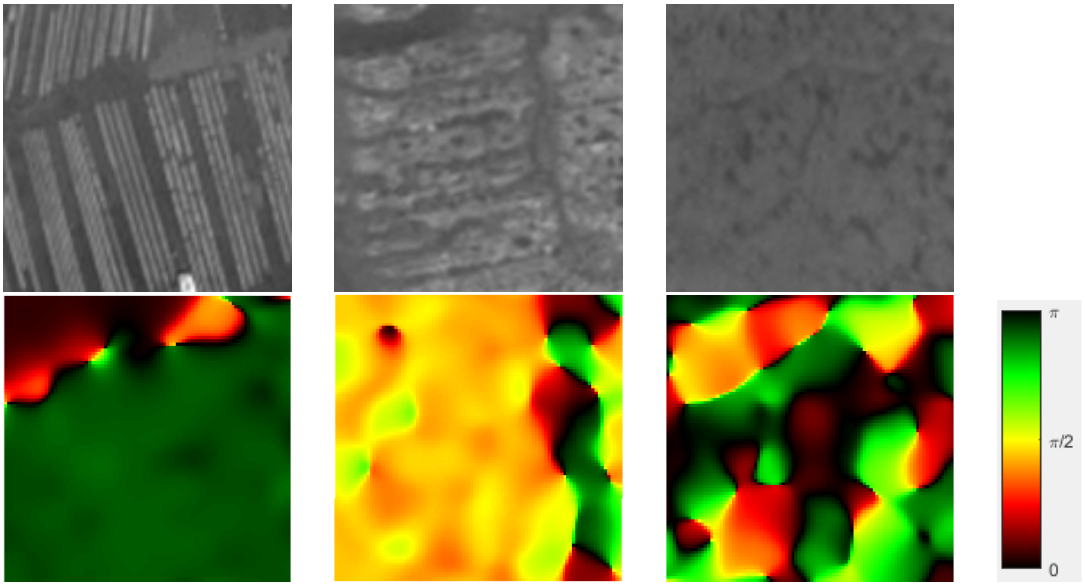


Figure 2: Exemples de textures de cultures de parcs ostréicoles et champs d'orientation associés.

moyennant le produit $\nabla I \nabla I^{\dagger}$ à l'aide d'un filtre convolutif gaussien W_T d'écart type σ_T :

$$\mathbf{Y} = W_T * \boldsymbol{\nabla} I \boldsymbol{\nabla} I^{\dagger} = W_T * \begin{bmatrix} I_x I_x & I_x I_y \\ I_x I_y & I_y I_y \end{bmatrix} \tag{5}$$

Le lissage spatial confère une certaine robustesse au bruit en permettant simultanément de paramétrer l'échelle à laquelle l'analyse de texture est menée. De manière générale, σ_T est choisi en fonction de la taille des motifs texturés présents dans l'image à traiter.

L'information géométrique extraite par le tenseur de structure peut être exploitée à travers sa diagonalisation. Le vecteur propre associé à la valeur propre la plus élevée indique l'orientation locale θ du gradient. La relation entre les deux valeurs propres fournit une mesure de l'anisotropie locale. Plus précisément, un large écart entre les valeurs propres est un indice de forte anisotropie locale. Par contre, dans le cas où les valeurs propres sont proches, l'anisotropie locale est faible. Une façon usuelle de mesurer l'anisotropie locale est de calculer le rapport entre leur différence et leur somme.

Les champs de TSL peuvent être représentés à travers des cartes scalaires de paramètres calculés en chaque point d'une image. Ces paramètres sont l'orientation, les valeurs propres, la mesure de l'anisotropie ou de l'énergie locale qui est la somme des valeurs propres. Un exemple de champ d'orientation est illustré dans la Figure 2. Par ailleurs, le champ de TSL peut également être représenté par un champ d'ellipses [Ennis 2006] dont les deux axes sont orientés selon les directions des vecteurs propres et sont de longueurs égales aux racines carrées des valeurs propres.

Les modèles statistiques définis sur les métriques AI and LE sont tout d'abord utilisés pour décrire des distributions marginales de TSL. L'ajustement des modèles théoriques aux distributions observées de TSL est évalué de manière expérimentale sur un ensemble de textures composées

d'un spectre assez large de motifs structuraux. Cette étude a montré des capacités descriptives supérieures pour les modèles log-euclidiens. Dans certains cas où des modèles de mélange des gaussiennes riemanniennes ont été nécessaires pour décrire la distribution observée de champs de TSL, une seule loi gaussienne multivariée dans l'espace LE s'est avérée suffisante pour caractériser la complexité d'information géométrique exprimée par le champs de tenseurs de structure. Ces résultats sont partiellement justifiés par un nombre de degrés de liberté supérieur des modèles LE par rapport aux modèles AI, pour un ordre de modèle identique dans les deux espaces.

Les modèles LE sont étendus pour décrire des distributions jointes de TSL dans l'objectif de caractériser à la fois les dépendances spatiales et multi-échelles au sein de champs de tenseurs de structure. Afin de caractériser les dépendances spatiales, des voisinages de tenseurs de structure sont considérés. Puis, des tenseurs étendus sont construits en chaque point en concaténant la représentation vectorielle LE correspondante à tous les tenseurs appartenant au voisinage considéré. Ainsi, en considérant un voisinage de p pixels, le tenseur étendu dans un point est représenté par un vecteur de dimension $3 \times p$. Un modèle gaussien multivarié $3 \times p$ ou un modèle de mélange de lois gaussiennes multivariées peuvent être utilisés par la suite pour caractériser le champ de tenseurs étendus. Afin de caractériser les dépendances multi-échelles, des champs de tenseurs sont tout d'abord calculés à plusieurs échelles d'analyse. Ceci correspond, en pratique, à calculer des champs de TSL pour différentes valeurs du paramètre σ_T. De manière similaire au cas de modélisation des dépendances spatiales, un TSL étendu est construit en chaque point en concaténant les formes vectorielles LE de TSL à la position considérée et à toutes les échelles d'analyse. Ensuite, les champs de tenseurs étendus peuvent être décrits par des modèles LE de dimensions supérieures.

Si les tenseurs étendus sont relativement simples à former, leur modélisation statistique est non-triviale. L'augmentation de la dimension de l'espace d'observation ainsi qu'un ensemble d'observation limité par la taille de l'image texturée et les effets de bords associés induisent des instabilités dans les processus d'inférence statistique.

L'invariance à la rotation est souvent une propriété requise pour les méthodes d'analyse de textures, notamment dans le cas des applications traitant des images texturées anisotropes. A cet effet, plusieurs stratégies sont proposées pour les méthodes paramétriques de modélisation de champs de tenseurs de structure. Ces stratégies s'articulent autour de deux axes : des méthodes basées sur l'estimation d'une tendance directionnelle globale et des méthodes reposant sur une recherche exhaustive, pour lesquelles la meilleure ressemblance entre deux modèles statistiques est recherchée en estimant les modèles statistiques sur des champs de tenseurs préalablement réorientés selon un ensemble prédéfini d'orientations. La deuxième stratégie est particulièrement adaptée dans les cas de textures pour lesquelles la notion d'orientation globale est ambiguë.

Application à l'analyse de textures. Procédés expérimentaux pour évaluer les potentialités des approches proposées.

Application en reconnaissance de texture

Les approches basées sur la modélisation statistique de TSL sont appliquées à la reconnaissance de textures, sur plusieurs bases de données regroupant des images de télédétection très haute résolution et des images de matériaux carbonés issues de la microscopie électronique à transmission haute résolution. Deux protocoles expérimentaux sont considérés : un protocole d'indexation et un protocole de classification supervisée.

Les approches LE et AI pour décrire les distributions marginales de TSL sont évaluées, dans un premier temps, dans un contexte d'indexation sur une base de données de parcs ostréicoles, composée de trois classes : cultures ostréicoles sur table (C_1), cultures ostréicoles en friche (C_2) et estran (C_3). Les textures dans les différentes classes sont caractérisées par différents degrés d'anisotropie, allant d'un fort caractère anisotrope (C_1) à une anisotropie très faible (C_3).

Pour chaque métrique, le champ de TSL est décrit par le barycentre (**M** - AI et $\boldsymbol{\mu}$ - LE), une loi gaussienne ($\mathbf{G_{AI}}$ et $\mathbf{G_{LE}}$) ainsi qu'un mélange de 3 lois gaussiennes ($\mathbf{3G_{AI}}$ et $\mathbf{3G_{LE}}$). Les performances des méthodes reposant sur la modélisation de champs de TSL sont comparées à celles obtenues par des méthodes de la littérature. A cet effet, des approches invariantes à la rotation de la matrice de co-occurrence des niveaux de gris [Haralick 1973, Haralick 1979] - **GLCM** - et du motif binaire local [Ojala 2002b] - **LBP** - sont utilisées ainsi que des méthodes basées sur la modélisation des distributions des coefficients de sous-bande obtenus par décomposition en ondelettes de l'image [Bombrun 2011b, Regniers 2014b, Regniers 2014c]. La méthode qui donne les meilleurs résultats est LBP. Les résultat d'indexation afférentes à toutes ces méthodes sont synthétisés dans le Tableau 1.

Pour tous les cas, les approches LE sont plus performantes que les approches AI, en termes de taux de reconnaissance et également en termes de temps de calcul. De plus, si sur l'espace AI, un mélange de lois gaussiennes s'avère nécessaire pour la reconnaissance, il en est autrement sur l'espace LE pour lequel une seule gaussienne est suffisante. De plus, les approches LE basées sur un modèle gaussien ou sur un modèle de mélange de lois gaussiennes se sont avérées plus performantes que les méthodes de la littérature testées sur cette base de données. Les mêmes observations ont été noté également sur d'autres bases de données utilisées pour la reconnaissance de textures. De plus, les résultats des approches basées sur la modélisation des champs de tenseurs se sont avérées peu dépendantes de la nature des textures traitées, tandis que les performances des méthodes de la littérature se sont révélés de qualité variable.

Sur l'espace LE, les modèles joints pour la caractérisation des dépendances spatiales au sein d'un champ de TSL ont amélioré légèrement les résultats par rapport aux modèles opérant sur des distributions marginales. La capacité intrinsèque des méthodes basées sur le tenseur de structure à

Table 1: Moyenne et écart type du taux de reconnaissance moyen en indexation, calculés sur 100 répétitions du protocole sur une base de données de parcs ostréicoles, pour l'ensemble des classes (TC) et pour chaque classe de textures C_1, C_2, C_3.

Méthode	TC	C_1	C_2	C_3
M	70.9± 3.7	70.1± 5.9	66.8± 4.4	75.7± 4.2
G_{AI}	73.3± 3.2	69.2± 5.4	72.8± 3.7	77.8± 4.8
$3G_{AI}$	77.1± 2.4	71.0± 4.7	78.8± 2.8	81.5± 4.6
μ	71.3± 3.2	71.1± 4.9	67.3± 4.0	75.6± 4.6
G_{LE}	79.3± 2.9	77.2± 5.8	77.9± 3.3	82.8± 4.6
$3G_{LE}$	79.8± 3.1	74.4± 5.5	80.9± 3.1	84.1± 4.7
GLCM	68.0± 3.1	52.4± 4.9	74.1± 3.3	77.5± 4.5
LBP	78.2± 2.7	75.6± 4.8	79.7± 3.4	79.2± 4.1
GCG	73.0± 3.1	74.9± 4.5	67.9± 4.0	76.3± 4.8

gérer l'invariance à la rotation a également été démontrés de manière expérimentale.

Application en synthèse de champ de tenseur de structure

Les capacités descriptives des modèles statistiques log-euclidiennes sont ensuite évaluées par des simulations de synthèse de champs de TSL. A cet effet, des approches mono-échelle ainsi que des approches pyramidales multi-échelles suivant une hypothèse markovienne sont proposées. Le principe d'une approche pyramidale consiste à synthétiser tout d'abord le champ de tenseur à une échelle basse résolution correspondante à l'étage le plus haut de la pyramide, et à ajouter ensuite, au fur et à mesure, des détails plus haute résolution en descendant les étages de la pyramide. Le résultat de synthèse obtenu à un étage donné de la pyramide est utilisé comme initialisation dans le processus de synthèse pour l'étage suivant. Deux approches multi-échelle sont proposées. Une première approche repose sur la modélisation de voisinages mono-échelle. Dans une deuxième approche, afin d'affecter un poids plus fort aux résultats de synthèse basse résolution, la modélisation de voisinages bi-échelle est proposée.

La synthèse repose sur des algorithmes de relaxation itératifs stochastiques (Metropolis et Metropolis - recuit simulé) et déterministe (ICM - *Iterated Conditional Modes*). La relaxation est effectuée pixel par pixel tandis que les parcours classiques lexicographique et aléatoire sont utilisés. Différents voisinages organisés en deux catégories sont également considérés : causale - lorsque toutes les positions du voisinage correspondent à des pixels qui ont déjà été synthétisés à l'itération courante - et non-causale, dans le cas contraire.

Les expériences sont effectuées sur des champs de TSL simulés ainsi que sur de champs de TSL calculés sur des textures réelles.

Dans le cas de champs de TSL simulés caractérisés par des motifs structuraux de petite taille, la synthèse aboutit à des résultats satisfaisants. De plus, les configurations des paramètres adaptés au motif structural étudié est généralement facile à identifier. Les expériences de synthèses menées sur

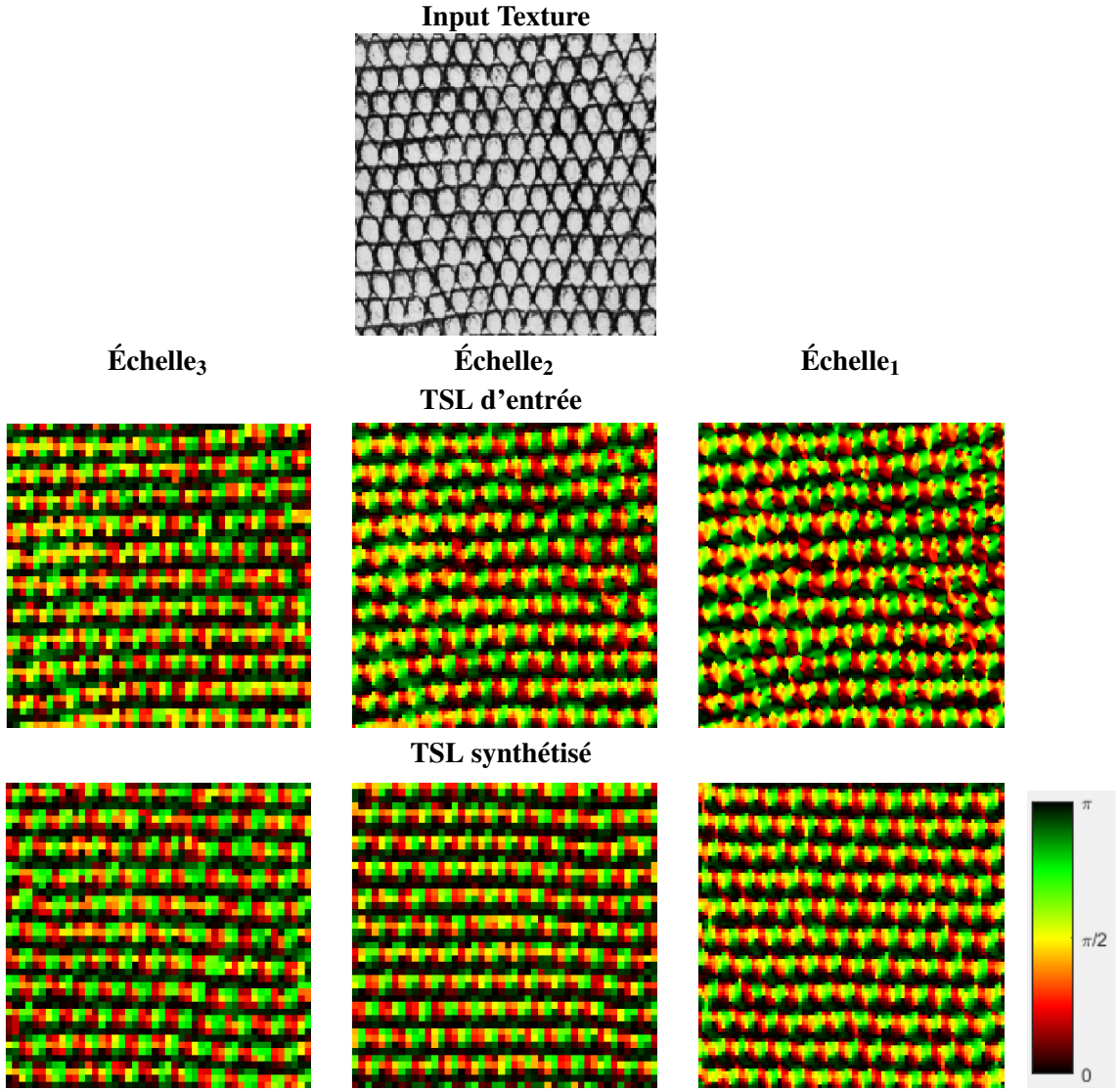


Figure 3: Exemple de résultats de synthèse de champs de TSL (représentés par des cartes d'orientation) pour une approche pyramidale reposant sur des voisinages mono-échelle.

des champs de TSL simulés, caractérisés par des motifs de taille plus grande, ont montré l'intérêt de l'utilisation des approches multi-échelle. Si la synthèse à une seule échelle n'aboutit pas, en général, à des résultats satisfaisants en utilisant des approches pyramidales, des configurations optimales de paramètres sont relativement simples à trouver pour obtenir une synthèse quasi parfaite de la structure du champ de TSL d'entrée.

Dans le cas des textures réelles composées de motifs structuraux de taille moyenne, les expériences ont montré un difficulté de synthèse à l'échelle native. Cela peut être dû au fait que des voisinages de taille large doivent alors être utilisés pour une caractérisation correcte du motif texturé. Étant donné qu'une augmentation de la taille du voisinage implique une augmentation de la dimension de l'espace d'observation, l'estimation de paramètres des modèles devient instable et, comme conséquence, les résultats de synthèse sont dégradés. Les approches de synthèse pyramidales, permettant à réduire la taille des voisinages utilisés, sont plus adaptés dans ces cas. Un exemple

de synthèse du champ de TSL d'une texture réelle, reposant sur une approche pyramidale à trois échelles, est illustré dans la Figure 3. Tout en montrant un potentiel réel de description des modèles proposées, les expériences menées montrent également une grande sensibilité aux choix de paramètres et en particulière au choix du voisinage.

Conclusion

Deux canevas paramétriques pour la description de champs de TSL calculés sur des textures réelles ont été proposés dans ce travail. Les méthodes considérées s'articulent autour de deux métriques riemanniennes, adaptées à la géométrie de l'espace de matrices symétriques semi-définies positives. Il s'agit, notamment, des métriques affine-invariante et log-euclidienne. Dans chacun des cas, un modèle de distribution gaussienne et de mélange associé ont été considérés pour une analyse statistique. Des méthodes d'estimation de leur paramètres sont proposées ainsi qu'une mesure de dissimilarité.

Les capacités de description des modèles statistiques proposés ont été évalués à travers deux applications.

Les approches basées sur la modélisation statistique de TSL ont tout d'abord été appliquées à la reconnaissance de textures, sur plusieurs bases de données regroupant des images de télédétection et des images de matériaux carbonés. Dans la plupart de cas, les approches proposées se sont avérées plus performantes que les méthodes de l'état de l'art. Les modèles LE opérant sur des distribution marginales ainsi que jointes de TSL se sont montrées les plus performantes en reconnaissance de textures, dans les contextes expérimentaux considérés . La capacité intrinsèque des approches basées sur le tenseur de structure à prendre en compte l'invariance à la rotation a également été éprouvée de manière expérimentale.

Les approches reposant sur la métrique LE ont également été appliquées à la synthèse de champs de TSL. A cet effet, des méthodes mono-échelle ainsi que des méthodes multi-échelle respectant une hypothèse markovienne ont été proposées. Les expériences de synthèse menées sur des champs de TSL à la fois simulés et calculés sur des textures réelles ont démontré un réel potentiel descriptif des modèles considérés. Ces expériences ont également montré une forte sensibilité au choix des paramètres qui peut s'expliquer par des instabilités d'estimation associées aux espaces de grandes dimensions.

Contents

Introduction

"The end is in the beginning and lies far ahead."

— Ralph Ellison, American Novelist

The local structure tensor (LST) [Bigün 1987, Knutsson 1989] is a commonly employed tool in computer vision for describing the structure of an image. Employed on non-textured images for tasks such as edge and corner detection [Harris 1988], the structure tensor is mainly used for describing the underlying geometry of textured images.

Computed in each pixel as the covariance matrix of the neighbouring gradients, the LST yields an estimation of the local orientation while associating a degree of coherence to the estimated orientation. While the LST has long been used for texture analysis tasks [Rousson 2003, Pham 2015b] or for texture synthesis [Akl 2015], its statistical modelling has rarely been addressed. For instance, a Von Mises distribution is proposed for modelling the LST orientations, in a joint segmentation-estimation context of a texture's orientation field [Da Costa 2012]. Another approach consists in employing the Kullback-Leibler divergence between two Gaussians as a dissimilarity measure between two tensors, considering the tensor as the covariance matrix of a normal law [de Luis-García 2008]. Apart from these few works, to the author's knowledge no detailed study of LST fields observed on real textures has been conducted in the literature, nor any experiments of statistical modelling of LST distributions. The objectives of this work are dual. A first goal is to investigate the mathematical strategies to model and handle LST distributions. A second goal consists in assessing the ability of such strategies to address the recognition and the synthesis of LST fields such as those observed on real textures.

As for mathematical modelling, the approaches must be addapted to the nature of LSTs that are defined as symmetric non-negative definite matrices. Thus, LSTs form a Riemannian manifold of particular curved shape. In consequence, as the tools of the Euclidean geometry do not account for the inherent curvature of the manifold, they are not adapted for handling LSTs. Similarly to the case of diffusion tensor imaging [Lenglet 2006, Dryden 2009], non-Euclidean tools and metrics adapted to the geometry of the LST space should be used instead.

Two complete Riemannian statistical frameworks for characterising covariance matrix sample

sets are thus introduced. They are defined on the affine-invariant (AI) and log-Euclidean (LE) metric spaces. Gaussian models are considered on both metric spaces, together with their mixture extensions. A Riemannian Gaussian model is considered on the AI space and a multivariate Gaussian distribution on the LE one. Solutions for parameter estimation are given and parametric dissimilarity measures between statistical models are proposed. The descriptive capacities of the two families of models are assessed and compared on randomly generated sample sets of covariance matrices.

The proposed methodology for LST characterisation relies on statistically modelling marginal and joint distributions of LSTs. Both AI and LE statistical models can be employed for characterising marginal distributions of LSTs. The models are evaluated and compared by analysing their fit to empirical LST fields computed on a set of textures composed of a wide diversity of textural patterns. Extensions of the LE statistical models are proposed in order to characterise joint distributions of LSTs thus allowing the characterisation of spatial and multiscale dependencies in LST fields.

Rotation invariance is a real prerequisite in applications dealing with anisotropic textures such as texture classification. Several strategies for ensuring rotation invariance are proposed for the LST based methods.

The potentials of AI and LE models are assessed in the context of texture recognition. Content based image retrieval and supervised classification experiments are performed on several datasets composed of textured and non textured images including remote sensing data and material images. The AI and LE statistical models for characterising marginal distributions of structure tensors are analysed and compared. On the LE metric space, statistical models for describing joint LST distributions are evaluated too. All the proposed methods are compared with state of the art texture analysis methods, in terms of classification accuracy, computational expenses and rotation invariance capacities.

The descriptive potential of the LE statistical models is further explored in the context of LST field synthesis. To this purpose, monoscale and multiscale pyramidal approaches for LST synthesis based on a Markovian hypothesis are developed. For the multiscale pyramidal approach, two methods are proposed. A first method consists in modelling an LST field's spatial dependencies at a single scale of analysis. A second method relies on statistically modelling spatial dependencies defined at two scales of analysis. All the proposed synthesis methods are tested on LST fields that are both artificially generated and estimated on real textures.

This thesis is organised in four chapters. The AI and LE statistical frameworks for characterising covariance matrix sample sets are described and analysed in Chapter 1. The notion of texture is introduced in Chapter 2, in addition to elements concerning the LST computation, decomposition and representation. The probabilistic and parametric methodology for characterising LST fields estimated on textured images is also described in this chapter. In Chapter 3, the performances of the LST AI and LE statistical modelling based approaches are evaluated in the context of texture recognition. The potential of the LE models for characterising spatial and multiscale dependencies in LST fields is further assessed in the context of LST synthesis, in Chapter 4.

Chapter 1

Statistical Modelling on the Space of Covariance Matrices

"The usual approach to science of constructing a mathematical model cannot answer the question of why there should be a universe for the model to describe. Why does the universe go to all the bother of existing ?"

–Stephen Hawking, Theoretical Physicist

Contents

1.1 Introduction

Covariance matrices are symmetric and positive semi-definite matrices of particular properties. An $m \times m$ real symmetric matrix, $\mathbf{Y}$, is positive semi-definite, if the following equations hold:

$$\mathbf{Y} - \mathbf{Y}^{\dagger} = 0 \tag{1.1}$$

and

$$\mathbf{x}^{\dagger}\mathbf{Y}\mathbf{x} \geq 0, \tag{1.2}$$

where † stands for the matrix transpose and $\mathbf{x} \neq 0$ is a column vector belonging to $\mathbb{R}^m$. Equation (1.2) implies that:

$$\det(\mathbf{Y}) \geq 0. \tag{1.3}$$

Namely, the determinant of $\mathbf{Y}$ is positive. In addition, the eigenvalues of $\mathbf{Y}$ are positive, as well.

For the special case when $m = 2$, the matrix $\mathbf{Y}$ can be written as follows:

$$\mathbf{Y} = \begin{pmatrix} \sigma_{xx} & \sigma_{xy} \\ \sigma_{xy} & \sigma_{yy} \end{pmatrix} \tag{1.4}$$

where the positive diagonal terms represent the variances of the x and y components of a random vector following a Gaussian distribution of covariance matrix $\mathbf{Y}$. The off diagonal terms are equal to each other and represent the covariance between the x and y components of the random vector. Equations (1.3) and (1.4) entail that:

$$|\sigma_{xy}| \leq \sigma_{xx}\sigma_{yy}. \tag{1.5}$$

Considering the afore-mentioned specific properties of a covariance matrix's elements, namely the positivity of the diagonal terms and the constraint on the off-diagonal components given by the inequality (1.5), the symmetric positive semi-definite matrices occupy only a subset of the vector space of symmetric matrices. The space they form (also referred to as a Riemannian manifold) has a particular curved shape. For a given matrix size, the space of symmetric positive semi-definite matrices forms a convex cone, with the inside comprising the symmetric strictly positive-definite matrices and with the singular positive semi-definite matrices residing on the cone's boundary, as illustrated in Figure 1.1, for 2×2 matrices. The matrix in this set having all eigenvalues equal to 0 is situated at the cone's origin.

While the tools of Euclidean geometry are well-suited for the characterisation of mathematical objects residing on flat spaces, they do not account for the inherent curvature of the geometrical space defined by the covariance matrices. As a consequence, the tools of the Euclidean geometry are not adapted for a complete characterisation of these matrices. Alternative Riemannian computation and statistical modelling frameworks should be used instead, that account for the specific geometry of the space defined by the symmetric positive matrices and lead, thus, to a faithful characterisation.

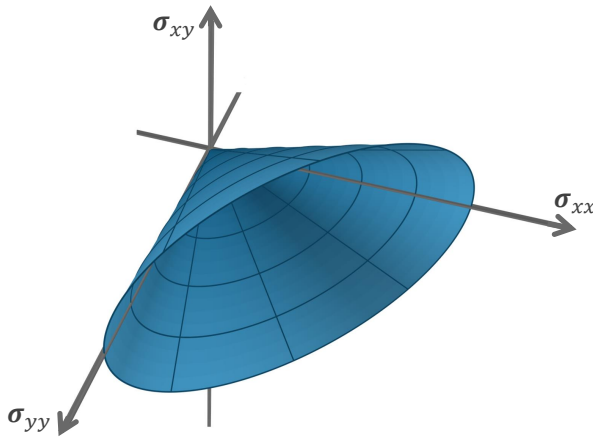


Figure 1.1: The positive semi-definite cone in $\mathbf{R}^3$ - the subspace occupied by symmetric positive semi-definite matrices in the vector space corresponding to 2×2 size symmetric matrices.

As previously explained, covariance matrices can be, by nature, either strictly definite positive (i.e. of strictly positive eigenvalues) or semi definite positive (i.e. of non-negative eigenvalues). However, to our knowledge, most of the works dealing with metrics and statistics adapted to covariance matrices focus on strictly positive definite matrices. They include the current chapter where two Riemannian statistical frameworks for covariance matrices characterisation and discrimination are presented. These statistical frameworks are limited to the case of symmetric positive definite (SPD) matrices. Adapted to the geometry of the manifold of covariance matrices, they overcome the inaccuracy and the defects arising from classical Euclidean computations on this space. These statistical frameworks are based on the affine-invariant and log-Euclidean metric, respectively, intrinsic Riemannian metrics with strong invariance properties. For each framework a distance measure between covariance matrices is proposed in addition to different statistical models for describing samples of covariance matrices. To be specific, a Gaussian distribution is proposed on each Riemannian statistical framework as well as their corresponding Gaussian mixture models, aimed to enrich the capabilities of the theoretical models to characterise the variability of experimental data on this manifold. Tractable parameter estimation algorithms are given for each statistical model. In addition to characterising covariance matrix distributions, one is often interested in their discrimination. To this purpose, solutions for measuring the dissimilarity between the estimated probability distributions of covariance matrix data samples are given for each of the proposed statistical models.

1.2 Related Work

While Euclidean based algorithms are typically successful when used on general symmetric matrices, this is no longer the case when dealing with SPD matrices. Euclidean computations do not account for the geometry of the SPD matrices manifold and may lead to mathematical discrepancies resulting in information and accuracy loss. A main limitation of the Euclidean framework is in

terms of its boundary. More precisely, Euclidean calculus on SPD matrices can lead to non-positive definite matrices as a result, which is a serious defect, as reported in the context of Diffusion Tensor Magnetic Resonance Imaging (DT-MRI) applications [Arsigny 2005, Arsigny 2006]. Instead, the computations performed on the Riemannian manifold of SPD matrices never lead to non-positive definite matrices. In addition, non-positive definite matrices should be at an infinite distance from the SPD ones.

In DT-MRI applications, the displacement of water molecules in biological tissues is locally encoded in the form of SPD matrices called diffusion tensors. The determinant of the diffusion tensors is an indicator of the dispersion associated to the local diffusion process [Arsigny 2006]. Namely, a large determinant is associated to a large dispersion in the local displacement of the water molecule and thus, to a strong diffusion process. Euclidean computations on diffusion tensors can give rise to an undesirable phenomenon called tensor swelling, as explained by Arsigny et al. in [Arsigny 2005, Arsigny 2006]. The tensor swelling phenomenon refers to the fact that the Euclidean mean between two diffusion tensors can have a grater determinant than the original determinants of the two tensors being averaged. In other words, the associated diffusion process can be strictly higher for the mean tensor than the diffusion processes corresponding to the original ones. This phenomenon is physically impossible. Tensor swelling occurs in tasks such as diffusion tensor interpolation, restoration or filtering of tensor-valued images.

Non-Euclidean alternatives for appropriate characterisation of SPD matrices are an active research topic in the literature. The statistical frameworks proposed in the different studies rely on various non-Euclidean metrics. A Riemannian metric specifies the geometry of a manifold $\mathcal{M}$. It consists of a smooth set of inner products g_p on the tangent space $T_p\mathcal{M}$ at each point p of the manifold [Lee 1997, Pennec 2006a]. One of the most popular Riemannian metrics is the affine-invariant one, called also Fisher-Rao [Atkinson 1981, Pennec 2006a]. The curvature of the manifold is taken into consideration and thus, the SPD space is accurately represented. In spite of providing precise and robust statistical representation tools on the space of SPD matrices, the curvature corrections induce very high computational costs. The log-Euclidean metric was proposed by Arsigny et al. in [Arsigny 2006], as an interesting alternative to the affine-invariant metric. It relies on a mapping of the SPD matrices in the logarithm domain. The LE mapping is a two steps process. First, the SPD matrices are applied the matrix logarithm yielding a symmetric matrix as a result. Second, the resulting symmetric matrix is expressed in vector-form. Thus, the manifold-valued data is transformed into SPD matrix logarithm vector-forms and the classical tools of the Euclidean geometry can be directly applied. Consequently, the computational burden of the affine-invariant metric is significantly reduced, while some of its theoretical properties are conserved [Arsigny 2006]. For both AI and LE metrics, the operations are defined directly on the manifold and do not rely on tangent space projections and on embedding the manifold into an ambient Euclidean space. AI and LE metrics both define true geodesic distances on the manifold of SPD matrices [Arsigny 2006, Said 2017, Jayasumana 2013].

Some other metrics are proposed in the literature on the manifold of SPD matrices [Dryden 2009, Jayasumana 2013]. In the objective of mean covariance matrix estimation, Dryden et al. [Dryden 2009] assess the properties of different non-Euclidean metrics in addition to the intrinsic Riemannian metrics AI and LE and to the Euclidean metric as well. The invariance properties of these distances are synthesised in Table 1.1. It shows that Rao's distance possesses the strongest invariance properties followed by the log-Euclidean one. Let $\mathbf{Y}_1$ and $\mathbf{Y}_2$ be two SPD matrices of size $m \times m$ and d a distance measure. The different invariance properties of the distance d are mathematically explained as follows:

- rotation and reflection invariance

$$d(\mathbf{Y}_1, \mathbf{Y}_2) = d(\mathbf{R}\mathbf{Y}_1\mathbf{R}^\dagger, \mathbf{R}\mathbf{Y}_2\mathbf{R}^\dagger), \tag{1.6}$$

  with $\mathbf{R}$ - $\in O(m)$ a rotation and reflection matrix, where $O(m)$ is the group of real orthogonal matrices of size $m \times m$ and $\dagger$ - the matrix transpose;

- scaling invariance

$$d(\mathbf{Y}_1, \mathbf{Y}_2) = d(\alpha\mathbf{Y}_1, \alpha\mathbf{Y}_2), \tag{1.7}$$

  with $\alpha > 0$;

- invariance under inversion

$$d(\mathbf{Y}_1, \mathbf{Id}) = d({\mathbf{Y}_1}^{-1}, \mathbf{Id}), \tag{1.8}$$

  where $\mathbf{Id}$ is the $m \times m$ identity matrix;

- invariance under affine transformations

$$d(\mathbf{Y}_1, \mathbf{Y}_2) = d(\mathbf{A}\mathbf{Y}_1\mathbf{A}^\dagger, \mathbf{A}\mathbf{Y}_2\mathbf{A}^\dagger), \tag{1.9}$$

  where $\mathbf{A}$ denotes a $m \times m$ real full rank matrix.

In addition to their invariance properties, the non-Euclidean distances have been analysed as well in terms of mean covariance matrix estimation for SPD matrix samples. While the mean computation is straightforward for most of the distances, in the case of Fisher Rao and Procrustes metrics, its estimation relies on iterative optimization algorithms, namely on the Riemannian gradient descent algorithm [Lenglet 2006] and on the generalized Procrustes algorithm, respectively [Gower 1975]. The performances of the different non-Euclidean metrics have been assessed in different experimental settings on both real and simulated SPD data samples. The experimental tasks consist of interpolation, *Principal Component Analysis* and mean covariance matrix estimation (in the case of simulated SPD data samples). The non-Euclidean metrics showed, globally, similar performances, as opposed to the Euclidean metric that performed poorly in all applications. The log-Euclidean,

Table 1.1: Invariance properties of non-Euclidean distances

Distance name	Invariance properties			
	Rotation and Reflection	Scaling	Inversion	Affine-Invariance
Euclidean	✓	-	-	-
Log-Euclidean	✓	✓	✓	-
Rao	✓	✓	✓	✓
Root-Euclidean	✓	-	-	-
Power-Euclidean	✓	-	-	-
Cholesky	-	-	-	-
Procrustes s&s*	✓	-	-	-
Full Procrustes s*	✓	✓	-	-

*Rao stands for the Riemannian distance induced by the Fisher-Rao metric, Procrustes s&s denotes the Procrustes size and shape distance while Full Procrustes s is the Full Procrustes shape distance.

root-Euclidean and Procrustes size and shape metrics lead to notably robust mean covariance matrix estimation in the particular case of samples close to rank defficient SPD matrices [Dryden 2009].

Different non-Euclidean metrics were exploited in order to define positive definite kernels on the Riemannian manifold for mapping SPD matrices to a higher dimensional Hilbert space where the tools of Euclidean geometry can be directly applied [Jayasumana 2013]. Several kernel based algorithms, namely *Principal Component Analysis*, *Support Vector Machine*, k-means and *Multiple Kernel Learning*, are extended to manifold-valued data. The superiority of the manifold kernel based approaches over their Euclidean counterparts as well as to the methods relying on tangent space projections is proven in several image processing SPD matrices applications such as segmentation, pedestrian detection, texture recognition and visual object categorisation. The non-Euclidean metrics investigated are: affine-invariant, log-Euclidean, power-Euclidean, Cholesky and root Stein divergence. The log-Euclidean one is prefered since it is the only one that in addition to defining a true geodesic distance on the manifold of SPD matrices it defines a positive definite kernel as well. In addition to its properties, the log-Euclidean metric outperforms all the other metrics in the context of visual objects categorisation [Jayasumana 2013].

While much attention has been given to appropriate metrics and distance measures on the Riemannian manifold of SPD matrices, fewer studies in the literature address the topic of adapted statistical models on the manifold of SPD matrices. The Wishart distribution is a well-known model for characterising SPD matrices [Wishart 1928, Goodman 1963]. Its shortcoming, however, is that it makes the hypothesis that the SPD matrices are covariance matrices of Gaussian vectors, which is rarely verified for real-world data applications. Some more recent propositions for covariance matrix statistical models are derived from the scalar product model. These models are particular in the sense that the observed covariance matrix is expressed as the product between a scaling factor τ and a scatter matrix $\boldsymbol{S}$ following a complex Wishart distribution. According to the choice of the prior probability of the scalar parameter τ, several different statistical models can be obtained. They include the $\mathcal{K}$ [Lee 1994] distribution, where τ follows the Gamma distribution and the KummerU

[Bombrun 2008] distribution where a Fisher distribution is considered as prior for the parameter τ. In spite of being successfully employed particularly in the context of SAR (Synthetic Aperture Radar) applications, their main shortcoming is that they are not adapted to the intrinsic geometry of the SPD data [Ilea 2017]. Propositions for alternative statistical models that overcome this drawback and that take into account the geometry of the space of covariance matrices have more recently been made in the literature. Gaussian distributions on the manifold of covariance matrices are proposed by Pennec [Pennec 2006a] and Lenglet et al. [Lenglet 2006] as well, both relying on the affine-invariant metric. While being well-adapted to manifold-valued data characterisation, these propositions are incomplete. They are based on asymptotic formulae and statistical inference is possible only in the particular case of compact distributions. These limitations are overcomed by Said et al. in [Said 2017]. As opposed to the previous propositions, the computation of the normalization constant of the probability density function is achieved, leading thus to an exact and tractable expression of the Riemannian Gaussian distribution. The distribution is parametrized by its central value given by the Riemannian center of mass and its dispersion around this central value. While an efficient indicator of the distribution's mean element, the Riemannian center of mass is less adapted for applications dealing with a significant amount of aberrant data due to its sensitivity to outliers [Bishop 2006]. This issue is addressed in [Hajri 2016], where a Riemannian Laplace distribution is proposed on the manifold of covariance matrices. Its central element is given by the Riemannian median [Yang 2010], more robust to the presence of outliers.

SPD matrices are mathematical objects with different physical interpretations, finding a wide range of applications in computer vision. For instance, the SPD matrices manifold is used in practice to represent covariance regions descriptors, employed for image processing tasks such as texture characterisation and classification [Tuzel 2006] or object detection and recognition [Tuzel 2008, Harandi 2012]. In medical imaging applications the SPD matrices manifold can be used to represent diffusion tensors that locally estimate the brain water molecules movement and encode the anatomical structures of cerebral white matter [Basser 1994, Pennec 2006b, Lenglet 2006, Arsigny 2006]. The manifold-valued data can be represented in practice by the structure tensors as well, which are mathematical tools used for characterising the local structure variations in textured images [de Luis-García 2008, Akl 2015, Rosu 2015a, Pham 2016a, Rosu 2016, Rosu 2017].

We focus in the following on the affine-invariant and log-Euclidean statistical frameworks. The choice for these two metric spaces is justified by their intrinsic Riemannian nature and by the strong invariance properties they possess compared to the non-Euclidean metrics previously reviewed. Thus, they provide the most faithful representations of manifold-valued data.

1.3 Affine-Invariant Model for Covariance Matrices

The focus of this section is on the affine-invariant models for the characterization of SPD matrix samples. A brief summary of the main properties of the AI metric space is made. Second, a complete

statistical framework for modelling SPD matrices is proposed, as an adequate alternative to classical Euclidean formulations. More precisely, a Riemannian Gaussian distribution defined on the space of covariance matrices and its corresponding mixture model are introduced. The research related to the proposition of the Riemannian Gaussian model has been carried out in the same research team and during the same time as the development of this thesis. The theoretical concepts on the Riemannian Gaussian model that are presented in this chapter (i.e. definition of the statistical model, parameter estimation, random sampling) are based on the work conducted by Said et. al [Said 2015, Said 2017, Said 2018, Rosu 2015a]. Furthermore, this statistical model in addition to its corresponding mixture model have been applied and assessed for the first time on samples of covariance matrices of real data in the context of texture discrimination during the development of this thesis [Rosu 2015a, Rosu 2017]. To this purpose, the problems of dissimilarity measurements between Riemannian Gaussian distributions and between mixtures of Riemannian Gaussian distributions as well have been addressed. The proposed solutions are presented at the end of this section.

1.3.1 The Affine-Invariant Metric Space

Let $\mathcal{P}_m$ be the space of all $m \times m$ real matrices that are symmetric and strictly positive definite. A matrix $\mathbf{Y} \in \mathcal{P}_m$, in addition to sharing the symmetry property with the set of symmetric positive semi-definite matrices, as given by equation (1.1), satisfies the following condition:

$$\mathbf{x}^\dagger \mathbf{Y} \mathbf{x} > 0, \tag{1.10}$$

$\forall \mathbf{x} \in \mathbb{R}^m$ with $\mathbf{x} \neq 0$. As a consequence

$$\det \mathbf{Y} > 0. \tag{1.11}$$

Namely, the determinant of an SPD matrix $\mathbf{Y}$ is strictly positive. The eigenvalues of $\mathbf{Y}$ are strictly positive as well.

The space $\mathcal{P}_m$ is a manifold of dimension $m(m+1)/2$. Each matrix $\mathbf{Y} \in \mathcal{P}_m$ can be expressed in terms of its polar coordinates given by the spectral decomposition:

$$\mathbf{Y} = \mathbf{R}^\dagger \, \mathrm{Diag}(\exp(r)) \mathbf{R} \tag{1.12}$$

, where $r = \{r_1, ...r_m\} \in \mathbb{R}^m$, $\mathrm{Diag}(\exp(r))$ is the diagonal matrix with the main diagonal entries $\{\exp(r_1), ... \exp(r_m)\}$ and $\mathbf{R}$ denotes an orthogonal matrix $\in O(m)$. For the special case when $m = 2$, the spectral decomposition $\mathbf{Y}$ can be expressed as a function of the triplet r_1, r_2 and θ, as follows:

$$\mathbf{Y} = \mathbf{Y}(r_1, r_2, \theta) = \begin{pmatrix} \cos(\theta) & \sin(\theta) \\ -\sin(\theta) & \cos(\theta) \end{pmatrix} \begin{pmatrix} \exp^{r_1} & 0 \\ 0 & \exp^{r_2} \end{pmatrix} \begin{pmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{pmatrix}. \tag{1.13}$$

The space $\mathcal{P}_m$ can be equipped with the Rao-Fisher Riemannian metric [Atkinson 1981]. It defines the squared length of a small displacement $\mathbf{dY}$ attached to the point $\mathbf{Y} \in \mathcal{P}_m$, as follows:

$$ds^2(\mathbf{Y}) = \mathrm{Tr}\left(\mathbf{Y}^{-1}\mathbf{dY}\right)^2, \tag{1.14}$$

where Tr denotes the trace.

The dissimilarity between two points $\mathbf{Y}_1, \mathbf{Y}_2 \in \mathcal{P}_m$ can be evaluated in terms of the distance defined by the Rao-Fisher metric. Namely, the geodesic distance or Rao's distance [Terras 1988]. The geodesic distance is equal to the length of the shortest curve connecting two points on the manifold P_m. Let $\mathbf{Y}_1, \mathbf{Y}_2 \in \mathcal{P}_m$ and $c : [0, 1] \to \mathcal{P}_m$ be a differentiable curve, with $c(0) = \mathbf{Y}_1$ and $c(1) = \mathbf{Y}_2$. The length $L(c)$ of the curve c is given by:

$$L(c) = \int_0^1 ds(c(t)) = \int_0^1 \left\| \frac{dc(t)}{dt} \right\| dt, \tag{1.15}$$

where $\|.\|$ denotes the Riemannian norm. Rao's distance $d(\mathbf{Y}_1, \mathbf{Y}_2), d : \mathcal{P}_m \times \mathcal{P}_m \to \mathbb{R}_+$, is the infimum of the length $L(c)$ with respect to all differentiable curves c. The Rao-Fisher metric transforms the space $\mathcal{P}_m$ into a Riemannian manifold with negative curvature. As a consequence, the infimum of $L(c)$ is attained by a unique curve γ:

$$\gamma(t) = \mathbf{Y}_1^{1/2}(\mathbf{Y}_1^{-1/2}\mathbf{Y}_2\mathbf{Y}_1^{-1/2})^t\mathbf{Y}_1^{1/2}, \tag{1.16}$$

known as the geodesic connecting the points $\mathbf{Y}_1$ and $\mathbf{Y}_2$. The length $L(\gamma)$ of the curve γ is the geodesic distance $d(\mathbf{Y}_1, \mathbf{Y}_2)$ [James 1973]:

$$d^2(\mathbf{Y}_1, \mathbf{Y}_2) = \mathrm{Tr}\left[\mathrm{logm}^2\left(\mathbf{Y}_1^{-1/2}\mathbf{Y}_2\mathbf{Y}_1^{-1/2}\right)\right] = \sum_{i=1}^{m} \log^2(\lambda_i), \tag{1.17}$$

where logm stands for the matrix logarithm function and λ_i is the i-th eigenvalue of the matrix $\mathbf{Y}_1^{-1/2}\mathbf{Y}_2\mathbf{Y}_1^{-1/2} \in \mathcal{P}_m$.

The Rao-Fisher metric endows the space $\mathcal{P}_m$ with affine-invariance properties [Pennec 2006b]. The most important properties from the perspective of the applications imagined in this work are recalled in the following.

A particular property of the space $\mathcal{P}_m$ is the fact that it is homogeneous under the action of $\mathrm{GL}(m)$, the group of $m \times m$ real and non-singular matrices. More precisely, for any $\mathbf{Y}_1, \mathbf{Y}_2 \in \mathcal{P}_m$, there exists a matrix $\mathbf{A} \in \mathrm{GL}(m)$ such that $\mathbf{Y}_2 = \mathbf{A}^\dagger\mathbf{Y}_1\mathbf{A}$. In addition, the homogeneity property also means that Rao-Fisher's metric and Rao's distance are invariant under the action of the group $\mathrm{GL}(m)$ on $\mathcal{P}_m$. In light of Rao's distance invariance properties, the following relations hold:

$$\begin{aligned} d(\mathbf{Y}_1, \mathbf{Y}_2) &= d(\mathbf{A}^\dagger\mathbf{Y}_1\mathbf{A}, \mathbf{A}^\dagger\mathbf{Y}_2\mathbf{A}) \\ &= d(\mathbf{Y}_1^{-1}, \mathbf{Y}_2^{-1}), \end{aligned} \tag{1.18}$$

for any $\mathbf{A} \in \mathrm{GL}(m)$. For more details on the geometric properties induced by the Rao-Fisher metric on the space $\mathcal{P}_m$, see [Said 2017, Said 2018].

1.3.2 Riemannian Gaussian Model

Said et al. proposed in [Said 2017] a Gaussian distribution on the Riemannian manifold of $\mathcal{P}_m$ matrices. A detailed and complete analysis of the statistical inference for the proposed statistical model is given as well. The generalisation of the Gaussian distribution to the Riemannian manifold has been considered before in the literature [Pennec 2006a, Lenglet 2006]. However, these propositions remain incomplete and difficult to apply in practice. They rely on approximations which are valid only in the particular cases of small variance distributions. These issues are overcome by Said et al. in [Said 2017], where an exact expression of a Riemannian Gaussian distribution is given for the first time in the literature.

The probability density of the Riemannian Gaussian distribution as well as the methods for parameters estimation are given in the following paragraphs. The Riemannian Gaussian model will later be employed on classification experiments for modelling covariance matrices (structure tensors) distributions of real-world textured images (see Chapter 3).

1.3.2.1 Definition

On the space $\mathcal{P}_m$ of SPD matrices, the probability density function of a Riemannian Gaussian distribution $\mathcal{G}(\mathbf{M}, \sigma)$ with respect to the Riemannian volume element associated to the affine-invariant metric is given by:

$$p(\mathbf{Y}|\mathbf{M}, \sigma) = \frac{1}{Z(\sigma)} \exp\left[-\frac{d^2(\mathbf{Y}, \mathbf{M})}{2\sigma^2}\right], \tag{1.19}$$

where $\mathbf{M} \in \mathcal{P}_m$ and $\sigma \in \mathbb{R}_+^*$ are the distribution's parameters, namely the central point represented here by the center of mass and a measure of the dispersion of the observations around this central point. As for $Z(\sigma)$, it is a normalizing constant depending uniquely on the dispersion σ and independent of the center of mass. Furthermore, d represents the geodesic distance defined in equation (1.17).

An exact expression of the normalising constant $Z(\sigma)$ is needed in order to define a Riemannian Gaussian distribution in terms of the probability density function given by equation (1.19). The normalising factor is given by:

$$Z(\sigma) = \int_{\mathcal{P}_m} \exp\left[-\frac{d^2(\mathbf{Y}, \mathbf{M})}{2\sigma^2}\right] dv(\mathbf{Y}), \tag{1.20}$$

where $dv(\mathbf{Y})$ is the Riemannian volume element. For the case of SPD matrices of size $m = 2$, its expression is given in closed form, as follows:

$$Z(\sigma) = (2\pi)^{3/2}\sigma^2 \exp\left(\frac{\sigma^2}{4}\right) \mathrm{erf}\left(\frac{\sigma}{2}\right), \tag{1.21}$$

where erf stands for the error function. For the case of larger SPD matrices of size $m > 2$, the normalising constant can be approximated by means of Monte Carlo integration [Ilea 2017].

An interesting invariance property of Rao's distance that will serve when using the Riemannian Gaussian model in later experiments, is the fact that for every $\mathbf{Y} \sim \mathcal{G}(\mathbf{M}, \sigma)$, $p(\mathbf{Y}|\mathbf{M}, \sigma) = p(\mathbf{Y}^*|\mathbf{Id}, \sigma)$ where $\mathbf{Y}^* = \mathbf{M}^{-\frac{1}{2}}\mathbf{Y}\mathbf{M}^{-\frac{1}{2}} \sim \mathcal{G}(\mathbf{Id}, \sigma)$ and $\mathbf{Id} \in \mathcal{P}_m$ is the identity matrix. $\mathcal{G}(\mathbf{Id}, \sigma)$ is called the *centred distribution*. For the special case of $m = 2$, by using the polar coordinates $\rho = r_1 - r_2$, $t = r_1 + r_2$ and θ, the probability density function of the centered distribution can be expressed as the product of the probability density functions of the 3 independent variables ρ, t and θ [Rosu 2015a]:

$$p(\mathbf{Y}|\mathbf{I}, \sigma) = p(\rho, t, \theta) \propto \exp\left(-\frac{\rho^2}{4\sigma^2}\right) \sinh\left(\frac{|\rho|}{2}\right) \times \exp\left(-\frac{t^2}{4\sigma^2}\right) \times \frac{1}{2\pi}, \tag{1.22}$$

where t is normally distributed and the orientation θ follows a uniform distribution.

1.3.2.2 Parameters Estimation

The two parameters of a Riemannian Gaussian distribution are estimated by means of the MLE (*Maximum Likelihood Estimation*) method. Let $\mathbf{Y} = \{\mathbf{Y}_1, ...\mathbf{Y}_N\}$ be a set of N independent and identically distributed (i.i.d.) samples of a Riemannian Gaussian distribution $\mathcal{G}(\mathbf{M}, \sigma)$. The log-likelihood function holds:

$$\begin{aligned} \mathcal{L}(\mathbf{Y}|\mathbf{M}, \sigma) &= \log \prod_{n=1}^{N} p(\mathbf{Y}_n|\mathbf{M}, \sigma) = \\ &= -N \log Z(\sigma) - \frac{1}{2\sigma^2} \sum_{n=1}^{N} d^2(\mathbf{Y}_n, \mathbf{M}), \end{aligned} \tag{1.23}$$

which leads to a maximum-likelihood estimate of $\mathbf{M}$ that corresponds to the Riemannian center of mass. Its estimate is obtained by minimizing the sum of squared distances between $\mathbf{M}$ and the observations $\mathbf{Y} = \{\mathbf{Y}_1, ...\mathbf{Y}_N\}$:

$$\widehat{\mathbf{M}} = \underset{\mathbf{M} \in \mathcal{P}_m}{\arg\max}\, \mathcal{L}(\mathbf{Y}|\mathbf{M}, \sigma) = \underset{\mathbf{M} \in \mathcal{P}_m}{\arg\min} \sum_{n=1}^{N} d^2(\mathbf{Y}_n, \mathbf{M}), \tag{1.24}$$

where $d(.)$ stands for the geodesic distance defined by equation (1.17). The solution to this minimization problem is given by means of a *Riemannian gradient descent* algorithm detailed in [Lenglet 2006].

The maximum-likelihood estimate of the dispersion parameter of a Riemannian Gaussian distribution is the solution $\widehat{\sigma}$ of a non-linear equation:

$$\frac{1}{N} \sum_{i=1}^{N} d^2(\mathbf{Y}_i, \mathbf{M}) = \sigma^3 \frac{d}{d\sigma} \log Z(\sigma) \tag{1.25}$$

solved by means of a *Newton-Raphson* algorithm.

1.3.3 Riemannian Gaussian Mixture Model

In this paragraph, a family of mixtures of Riemannian Gaussian distributions on the space $\mathcal{P}_m$ is introduced. Mixture models are generally considered in order to enrich the ability of theoretical statistical models to describe distributions of real data. Thus, if the number of mixture components is well chosen, the statistical model should normally be well-suited to encompass the variability that arises in real-world data and to accurately describe its distribution.

The probability density function of a mixture of K Riemannian Gaussian distributions is given by:

$$p(\mathbf{Y}|(\omega_k, \mathbf{M}_k, \sigma_k)_{k=1,...,K}) = \sum_{k=1}^{K} \omega_k \, p(\mathbf{Y}|\mathbf{M}_k, \sigma_k), \tag{1.26}$$

where $\omega_k > 0$ are the weights of sum 1 associated to each distribution $k, k = 1, ...K$ in the mixture model. In addition, $\mathbf{M}_k$ and σ_k are the center of mass and, respectively, the dispersion of the k-th Riemannian Gaussian distribution in the mixture model. The parameters: $\omega_k, \bar{\mathbf{Y}}_k$ and σ_k are estimated by employing an adaptation of the *Expectation-Maximization* (EM) algorithm to the Riemannian geometry of the space $\mathcal{P}_m$, proposed by Said et al. in [Said 2015] and applied in [Rosu 2015a]. Details on the algorithm in the specific case of a Riemannian Gaussian mixture model are given in Appendix C.

1.3.4 Affine-Invariant Model - Random Sampling

A method for sampling from a Riemannian Gaussian distribution of dimension m was developed in [Said 2017] and previously implemented in [Rosu 2015a]. In the following we will briefly describe the steps that need to be followed in the special case when $m = 2$. Given the invariance properties of the space $\mathcal{P}_2$ equipped with the AI metric (see Section 1.4.1), generating a sample set $\mathbf{Y} = \{\mathbf{Y}_1, ...\mathbf{Y}_N\}$ of SPD matrices of distribution $\mathcal{G}(\mathbf{M}, \sigma)$ can be done by firstly drawing a sample set $\mathbf{Y}^* = \{\mathbf{Y}_1^*, ...\mathbf{Y}_N^*\}$ from the centered distribution $\mathcal{G}(\mathbf{Id}, \sigma)$ and by further centering the random data samples $\mathbf{Y}_n^*, i = 1, ...N$ around the center of mass, as follows: $\mathbf{Y}_n = \mathbf{M}^{1/2}\mathbf{Y}_n^*\mathbf{M}^{1/2}$. Sampling from the distribution $\mathcal{G}(\mathbf{Id}, \sigma)$ can be done in two steps:

- Considering the expression of the Riemannian Gaussian distribution in terms of polar coordinates (see equation (1.22)), sampling from $\mathcal{G}(\mathbf{Id}, \sigma)$ is equivalent to sampling from the probability densities of $\mathbf{R}$ and $r = (r_1, r_2)$ (see equations (1.12) and (1.13)). The matrix $\mathbf{R}$ is sampled from a uniform distribution on the space O_2 of 2×2 orthogonal matrices, i. e. uniform angle $\theta \in [0, \pi]$. The probability density function of r can be decomposed into the product of the probability density functions of two independent variables, $\rho = r_1 + r_2$ and $t = r_1 - r_2$. Consequently, in this specific case, sampling from the density of r can be achieved by sampling from univariate probability densities, given in equation (1.27):

$$
\begin{aligned}
p(r) &\propto p(t) \times p(\rho) \\
p(\rho) &\propto \exp\left(-\frac{\rho^2}{4\sigma^2}\right) \sinh\left(\frac{|\,\rho\,|}{2}\right) \\
p(t) &\propto \exp\left(-\frac{t^2}{4\sigma^2}\right);
\end{aligned}
\tag{1.27}
$$

While t appears to be normally distributed and thus easy to sample, ρ can be sampled by employing the Metropolis-Hastings technique - a Monte Carlo Markov Chain based sampling method [Metropolis 1953, Hastings 1970].

- The sample set $\{\mathbf{Y}_1^*, ...\mathbf{Y}_N^*\}$ is obtained by equation (1.13), starting from the previously generated sample sets of polar coordinates, namely $t = t_1, ...t_N, \rho = \rho_1, ...\rho_N$ and $\theta = \theta_1, ...\theta_N$, as follows: $\mathbf{Y}_n^* = \mathbf{Y}_n^*((t_n + \rho_n)/2, (t_n - \rho_n)/2, \theta_n)$, $n = 1, ...N$.

When it comes to generating a data sample set of size N from a mixture model of K distributions, the task breaks down to sampling from each individual distribution in the mixture model. More precisely, in the case of a Riemannian Gaussian mixture model, for each $k = 1, ...K$, a sample set of size $N_k = \lceil \omega_k N \rceil$ is drawn from its corresponding Riemannian Gaussian distribution of parameters $\mathbf{M}_k$ and σ_k, where $\lceil\,\rceil$ stands for the ceiling operation. The data sample of the mixture model is obtained by concatenating all the samples thus generated from the K individual Riemannian Gaussian distributions.

1.3.5 Dissimilarity Measures between AI Models

When dealing with collections of data, in addition to their characterisation by means of different statistical models, one is often interested in comparing different data sets. This task of data set discrimination can be done by comparing the statistical models describing them. In this section we address precisely the topic of dissimilarity measures between the different Riemannian statistical models proposed so far for describing data samples of SPD matrices.

Let us first consider two samples of data drawn from two probability distributions of unknown parameters, denoted by p and q. One way of comparing these data samples is to assess the dissimilarity between the probability distributions $\hat{p}$ and $\hat{q}$ estimated from the respective data samples. A popular choice for measuring the dissimilarity between two probabilistic models is the Kullback-Leibler's divergence [Kullback 1951]:

$$
D_{KL}(\hat{p}, \hat{q}) = \int_x \hat{p}(x) \log \frac{\hat{p}(x)}{\hat{q}(x)} dx. \tag{1.28}
$$

Given the asymmetrical nature of the Kullback-Leibler's divergence, it is quite common to consider

its symmetric version, called the Jeffrey divergence, as follows:

$$D_J(\hat{p}, \hat{q}) = D_{KL}(\hat{p}, \hat{q}) + D_{KL}(\hat{q}, \hat{p}). \tag{1.29}$$

Kullback-Leibler's divergence (KLD) is specific to each probability density and it presents the advantage of versatility. Indeed, it can be employed for every probability model, regardless its parametrisation. However, for some statistical models there is no analytically tractable formulation of the divergence, as is the case of the Riemannian Gaussian distributions.

In the absence of an analytical expression of the KLD between two statistical models $\hat{p}$ and $\hat{q}$, the divergence can be estimated by means of Monte Carlo sampling techniques [Hershey 2007, Kwitt 2009], as follows:

$$\hat{D}_J(\hat{p}, \hat{q}) = \frac{1}{N}\sum_{n=1}^{N} \log \frac{\hat{p}(x_n)}{\hat{q}(x_n)} + \frac{1}{N}\sum_{n=1}^{N} \log \frac{\hat{q}(y_n)}{\hat{p}(y_n)}, \tag{1.30}$$

where $\{x_1, ...x_N\}$ and $\{y_1, ...y_N\}$ represent two data samples generated from the estimated probability distributions $\hat{p}$ and $\hat{q}$. Thus, a firs step when approximating the KLD between two distributions is to generate a set of N samples from their respective probability densities. Generating random samples from a Riemannian Gaussian distribution will be done according to the steps previously described in Section 1.3.4.

Details upon the evaluation of the Jeffrey divergence in the case of each of the affine-invariant statistical models presented here are given in the following.

1.3.5.1 Riemannian Gaussian Models

When comparing two data samples, each modelled by a Riemannian Gaussian distribution, as the Kullback-Leibler divergence is not given in analytical form, it will be estimated by Monte Carlo sampling techniques, as follows.

Let $\mathbf{Y}_1 = \{\mathbf{Y}_{11}, ...\mathbf{Y}_{1N}\}$ and $\mathbf{Y}_2 = \{\mathbf{Y}_{21}, ...\mathbf{Y}_{2N}\}$ be two samples of SPD matrices generated from 2 Riemannian Gaussian distributions, $\mathcal{G}(\widehat{\mathbf{M}}_1, \widehat{\sigma}_1)$ and $\mathcal{G}(\widehat{\mathbf{M}}_2, \widehat{\sigma}_2)$, denoted by $\hat{p}$, $\hat{q}$. By evaluating the expression given in equation (1.30), the estimation of the Jeffrey divergence between two SPD matrix samples issued from two Riemannian Gaussian distributions is of the following form:

$$\hat{D}_J(\hat{p}, \hat{q}) = \frac{1}{N}\sum_{n=1}^{N}\left[\frac{d^2(\mathbf{Y}_{1n}, \widehat{\mathbf{M}}_2)}{2\widehat{\sigma}_2^2} - \frac{d^2(\mathbf{Y}_{1n}, \widehat{\mathbf{M}}_1)}{2\widehat{\sigma}_1^2} + \frac{d^2(\mathbf{Y}_{2n}, \widehat{\mathbf{M}}_1)}{2\widehat{\sigma}_1^2} - \frac{d^2(\mathbf{Y}_{2n}, \widehat{\mathbf{M}}_2)}{2\widehat{\sigma}_2^2}\right], \tag{1.31}$$

where d denotes the geodesic distance, defined in equation (1.17). As a remark, the normalising factors $Z(\sigma_1)$ and $Z(\sigma_2)$ simplify during developments. The independence of the approximation of the Jeffrey divergence given by equation (1.31) between two Riemannian Gaussian models with respect of the normalising factors of the two distributions has positive consequences on the computation time.

1.3.5.2 Riemannian Gaussian Mixture Models

The dissimilarity measure between two mixture models of Gaussian Riemannian distributions, assessed in terms of Jeffrey divergence, is also approximated by means of Monte Carlo sampling. Further details specific to this particular case are given in the following.

Let $\mathbf{Y}_1 = \{\mathbf{Y}_{11}, ...\mathbf{Y}_{1N}\}$ and $\mathbf{Y}_2 = \{\mathbf{Y}_{21}, ...\mathbf{Y}_{2N}\}$ be two samples of SPD matrices generated from 2 mixture models of K Riemannian Gaussian distributions: $\hat{p}$ of parameters $\widehat{\omega}_{1k}, \widehat{\mathbf{M}}_{1k}, \widehat{\mathbf{\Sigma}}_{1k}$ and $\hat{q}$ of parameters $\widehat{\omega}_{2k}, \widehat{\mathbf{M}}_{2k}, \widehat{\mathbf{\Sigma}}_{2k}$, with $k = 1...K$. The probability density functions for the data in the sample set $\mathbf{Y}_1$ are given by:

$$\begin{aligned} \hat{p}(\mathbf{Y}_{1n}) &= \sum_{k=1}^{K} \widehat{\omega}_{1k} \hat{p}_k(\widehat{\mathbf{M}}_{1k}, \widehat{\sigma}_{1k}) \\ \hat{q}(\mathbf{Y}_{1n}) &= \sum_{k=1}^{K} \widehat{\omega}_{2k} \hat{q}_k(\widehat{\mathbf{M}}_{2k}, \widehat{\sigma}_{2k}), \end{aligned} \tag{1.32}$$

where $n = 1, ...N$, $\hat{p}_k \sim \mathcal{G}(\widehat{\mathbf{M}}_{1k}, \widehat{\sigma}_{1k})$ and $\hat{q}_k \sim \mathcal{G}(\widehat{\mathbf{M}}_{2k}, \widehat{\sigma}_{2k})$. Identical relations hold for the data in the sample set $\mathbf{Y}_2$.

By developing the expression in equation (1.30), the approximation of the Jeffrey divergence between two Riemannian Gaussian mixture distributions $\hat{p}$ and $\hat{q}$ is as follows:

$$\begin{aligned} \hat{D}_J(\hat{p}, \hat{q}) = \frac{1}{N} \sum_{n=1}^{N} \Bigg\{ & \log \sum_{k=1}^{K} \widehat{\omega}_{1k} Z^{-1}(\widehat{\sigma}_{1k}) \exp\left[-\frac{d^2(\mathbf{Y}_{1n}, \widehat{\mathbf{M}}_{1k})}{2\widehat{\sigma}_{1k}^2} \right] - \\ & - \log \sum_{k=1}^{K} \widehat{\omega}_{2k} Z^{-1}(\widehat{\sigma}_{2k}) \exp\left[-\frac{d^2(\mathbf{Y}_{1n}, \widehat{\mathbf{M}}_{2k})}{2\widehat{\sigma}_{2k}^2} \right] + \\ & + \log \sum_{k=1}^{K} \widehat{\omega}_{2k} Z^{-1}(\widehat{\sigma}_{2k}) \exp\left[-\frac{d^2(\mathbf{Y}_{2n}, \widehat{\mathbf{M}}_{2k})}{2\widehat{\sigma}_{2k}^2} \right] - \\ & - \log \sum_{k=1}^{K} \widehat{\omega}_{1k} Z^{-1}(\widehat{\sigma}_{1k}) \exp\left[-\frac{d^2(\mathbf{Y}_{2n}, \widehat{\mathbf{M}}_{1k})}{2\widehat{\sigma}_{1k}^2} \right] \Bigg\}, \end{aligned} \tag{1.33}$$

where $Z(\sigma_{1k})$ and $Z(\sigma_{2k})$ are the normalization constants of the two Riemannian Gaussian distributions, defined in equation (1.20) and given in closed form in the case when the SPD matrices are of size 2×2 (see equation (1.21)). In addition, d stands for the geodesic distance, given in equation (1.17).

The affine-invariant metric is an intrinsic Riemannian metric and thus, the statistical models defined on this metric space are robust and provide an accurate characterisation of SPD matrix samples. In spite of its strong theoretical properties, this statistical framework relies on complex and recursive algorithms of high computational burden. This time efficiency problem arises both when estimating the mean of an SPD matrix sample set and when estimating the parameters of a Riemannian Gaussian distribution or of its corresponding mixture model. In addition, when assessing

the dissimilarity between two affine-invariant statistical models, the Kullback-Leibler's divergence is not available in analytical form. For these cases the Kulback-Leiblers's divergence is approximated by means of Monte Carlo sampling which induces significantly higher computational costs as opposed to the case when its expression is given in closed form. As a consequence, the execution time in real applications dealing with SPD data characterisation and discrimination becomes almost prohibitive.

1.4 Log-Euclidean Model for Covariance Matrices

The second Riemannian statistical framework on the space of covariance matrices considered in this work is based on the log-Euclidean metric. Similar to the AI metric, it aims to overcome the limitations of the Euclidean computation on the space of covariance matrices. A brief introduction to the LE metric space as well as a distance measure and a complete statistical framework are proposed in the following paragraphs. A multivariate Gaussian model along with its corresponding multivariate Gaussian mixture model are proposed for describing samples of covariance matrices on the LE metric space. Solutions for measuring the dissimilarity between LE multivariate Gaussian models and between mixtures of multivariate Gaussian models are given in the last part of this section.

1.4.1 The Log-Euclidean Metric Space

As previously presented, the affine-invariant metric space is equipped with well-adapted properties for performing mathematical operations and statistical modelling on the covariance matrices manifold. In spite of providing a robust and accurate representation and computation framework for covariance matrices, it relies on complex algorithms inducing a high computational burden. An interesting alternative to the AI representation and computation tools for covariance matrices, originally proposed by Arsigny et al. in [Arsigny 2006] is offered by the log-Euclidean metric space. As stated in [Arsigny 2006, Arsigny 2005], while the LE metric is endowed with the same excellent theoretical properties as the AI metric, its mathematical operations are much simplified and the computational cost is significantly reduced as well.

The LE metric provides a vector space representation of covariance matrices by mapping the covariance matrices in the logarithm domain. The mathematical operations on covariance matrices become, thus, Euclidean in the logarithm domain [Arsigny 2006]. Consequently, the complexity and computational expenses of the algorithms residing on this metric space are significantly reduced. As opposed to computations based on the AI metric, the LE based ones do not use curvature corrections. In spite of this consistent simplification, the main properties of the Riemannian geometry represention framework for covariance matrices are preserved by the LE metric [Arsigny 2005, Arsigny 2006]. While the LE metric space does not provide full affine-invarice properties, it provides similarity-invariance properties [Arsigny 2006] that consist of invariance to rotation, reflection, scaling and inversion (see Table 1.1). However, this slight loss in the precision

of representation did not affect the covariance matrix computations and statistical modelling performed in this work, as it will be lately shown by the different experiments proposed in this report (see Section 3.5.1).

The LE metric has been tested against the Euclidean and affine-invariant metrics as well, in different applications. Some examples are regularisation and interpolation [Arsigny 2006] or resampling, dense extrapolation of sparse data and anisotropic filtering [Arsigny 2005] of both synthetic and clinical 3D Diffusion Tensor Imaging (DTI) data. All these experiments showed that as in the case of the AI metric, the LE metric based computations successfully overcome the defects of Euclidean calculus. While it yields similar experimental results to the AI metric, the algorithms based on the LE metric proved to be at least 6 times faster in the applications enlisted before. The only detectable difference in these applications on tensor data according to [Arsigny 2005], is that the LE metric space algorithms generate slightly more anisotropic tensors as results.

For yielding the vector space representation of covariance matrices, the later need to be mapped on the LE space. This mapping consists of computing the matrix logarithm $\mathbf{Y}_{\mathrm{LE}}$ of the covariance matrix $\mathbf{Y}$:

$$\mathbf{Y}_{\mathrm{LE}} = \mathrm{logm}(\mathbf{Y}). \tag{1.34}$$

As in the scalar case, the matrix logarithm is defined as the inverse of the matrix exponential. For any square matrix $\mathbf{A}$, the matrix exponential can be computed as follows [Arsigny 2006]:

$$\mathrm{expm}(\mathbf{A}) = \sum_{k=0}^{\infty} \frac{\mathbf{A}^k}{k!}. \tag{1.35}$$

When it comes to the matrix logarithm, it should be noted that neither its existence nor its uniqueness is guaranteed in the case of general invertible matrices. Nevertheless, in the case of a matrix $\mathbf{A}$ that is sufficiently close to the identity, its matrix logarithm can be computed by means of an infinite power series [Stillwell 2008]:

$$\mathrm{logm}(\mathbf{A}) = \sum_{k=1}^{\infty} (-1)^{k+1} \frac{(\mathbf{A} - \mathbf{Id})^k}{k}, \tag{1.36}$$

whose convergence is guaranteed for $\|\mathbf{A} - \mathbf{Id}\| < 1$, where $\|.\|$ stands for the Frobenius matrix norm. This non-trivial computation is simplified in the case of symmetric matrices. However, this work concerns uniquely SPD matrices. In consequence, uniquely the matrix logarithm computation of an SPD matrix $\mathbf{Y}$ will be addressed in the following. In this particular case, the matrix logarithm can be obtained in two simple steps. The first step consists of diagonalizing the matrix, as follows:

$$\mathbf{Y} = \mathbf{R}\mathbf{\Lambda}\mathbf{R}^{\dagger}, \tag{1.37}$$

where $\mathbf{R}$ is a rotation matrix and $\mathbf{\Lambda}$ is the diagonal matrix of the eigenvalues of $\mathbf{Y}$. In the 2×2 case it holds:

$$\mathbf{\Lambda} = \begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix}, \text{with} \lambda_1 \geq \lambda_2, \text{ and } \lambda_1, \lambda_2 > 0. \tag{1.38}$$

The second step consists of computing the matrix logarithm itself:

$$\operatorname{logm} \mathbf{Y} = \operatorname{logm}(\mathbf{R}\mathbf{\Lambda}\mathbf{R}^{\dagger}). \tag{1.39}$$

Given the matrix logarithm properties of invertible matrices:

$$\operatorname{logm}(\mathbf{R}\mathbf{\Lambda}\mathbf{R}^{-1}) = \mathbf{R} \operatorname{logm}(\mathbf{\Lambda})\mathbf{R}^{-1} \tag{1.40}$$

and the following property of orthogonal matrices:

$$\mathbf{R}^{-1} = \mathbf{R}^{\dagger}, \tag{1.41}$$

the expression in equation (1.39) becomes:

$$\operatorname{logm}(\mathbf{Y}) = \mathbf{R} \operatorname{logm}(\mathbf{\Lambda})\mathbf{R}^{\dagger}. \tag{1.42}$$

Thus, the matrix logarithm is easily computed by taking the scalar logarithm of each eigenvalue in the matrix $\mathbf{\Lambda}$ and recomposing its equivalent covariance matrix. In the 2×2 case:

$$\mathbf{Y}_{\mathrm{LE}} = \mathbf{R} \begin{pmatrix} \log \lambda_1 & 0 \\ 0 & \log \lambda_2 \end{pmatrix} \mathbf{R}^{\dagger} = \begin{pmatrix} y_{\mathrm{LE}_{xx}} & y_{\mathrm{LE}_{xy}} \\ y_{\mathrm{LE}_{xy}} & y_{\mathrm{LE}_{yy}} \end{pmatrix}. \tag{1.43}$$

It should be noted that the eigenvalues λ_1 and λ_2 are equal to the terms $\exp^{r_1}$ and $\exp^{r_2}$ of the diagonal matrix given previously by equation (1.12). When expressing a covariance matrix in terms of its spectral decomposition, the eigenvalues are typically used as parameters for the diagonal matrix. A different parametrisation has been preferred in the previous case of affine-invariant model in order to facilitate the presentation of the Riemannian Gaussian distribution. More precisely, its probability density function can be decomposed into the product of the probability density functions of 3 other variables out of which two can be expressed in terms of r_1 and r_2 (see equation (1.22)).

The matrix logarithm of an SPD matrix always results in a symmetric matrix $\mathbf{Y}_{\mathrm{LE}}$ that can be written as well in vector form, as follows:

$$\vec{\mathbf{Y}}_{\mathrm{LE}} = \left[y_{\mathrm{LE}_{xx}}, \sqrt{2} y_{\mathrm{LE}_{xy}}, y_{\mathrm{LE}_{yy}} \right]^{\dagger}. \tag{1.44}$$

Weighting the off-diagonal term with $\sqrt{2}$ ensures that:

$$\|\vec{\mathbf{Y}}_{\mathrm{LE}}\| = \|\mathbf{Y}_{\mathrm{LE}}\|, \tag{1.45}$$

namely that the Euclidean norm of the vector-form representation is equal to the Frobenius norm of the matrix representation. The matrix logarithm mapping yields, thus, a vector form representation of the space of SPD matrices in the logarithm domain. The inverse mapping operation from the LE vector space back to the covariance matrix space is simply done by computing the matrix exponential [Arsigny 2006]. The matrix exponential of a symmetric matrix always yields an SPD matrix, so there is a one to one correspondance between symmetric matrices and SPD ones, under the matrix exponential operation [Arsigny 2005].

Once the covariance matrices mapped on the LE metric space, all the standard mathematical rules and operations of the classical Euclidean geometry apply, without distorting the covariance matrix representations. Thus, the computational complexity associated to the affine-invariant metric space is significantly reduced while the strong invariance properties inherent to a Riemannian metric are preserved.

Distance between Covariance Matrices

The distance between two covariance matrices mapped on the LE metric space $\mathbf{Y}_{\mathrm{LE}_1}$ and $\mathbf{Y}_{\mathrm{LE}_2}$ can be assessed in terms of the Frobenius distance:

$$d_F(\mathbf{Y}_{\mathrm{LE}_1}, \mathbf{Y}_{\mathrm{LE}_2}) = \sqrt{Tr((\mathbf{Y}_{\mathrm{LE}_1} - \mathbf{Y}_{\mathrm{LE}_2})(\mathbf{Y}_{\mathrm{LE}_1} - \mathbf{Y}_{\mathrm{LE}_2})^{\dagger})}. \tag{1.46}$$

The Frobenius distance between the matrices $\mathbf{Y}_{\mathrm{LE}_1}$ and $\mathbf{Y}_{\mathrm{LE}_2}$ of size 2×2 is, in fact, equivalent to the Euclidean distance between their 3 component vector representations $\vec{\mathbf{Y}}_{\mathrm{LE}_1}$ and $\vec{\mathbf{Y}}_{\mathrm{LE}_2}$ (see equation (1.44)) given by:

$$d^2(\vec{\mathbf{Y}}_{\mathrm{LE}_1}, \vec{\mathbf{Y}}_{\mathrm{LE}_2}) = \sum_{i=1}^{3} [\vec{\mathbf{Y}}_{\mathrm{LE}_1}(i) - \vec{\mathbf{Y}}_{\mathrm{LE}_2}(i)]^2. \tag{1.47}$$

For simplicity reasons, the vector form representation $\vec{\mathbf{Y}}_{\mathrm{LE}}$ of the covariance matrix will be denoted from now on by $\mathbf{Y}_{\mathrm{LE}}$.

1.4.2 Multivariate Gaussian Model

In the following paragraphs, a multivariate Gaussian distribution is proposed for characterising samples of LE mapped covariance matrices. This statistical model was previously evoked by Arsigny in [Arsigny 2006] as a potential way to represent LE mapped covariance matrices, but, to our knowledge, never really used in practice. The probability density function and the parameter estimates are given in the following. This model will later be experimentally applied for characterising samples of covariance matrices of real textured image data in Chapters 3 and 4.

1.4.2.1 Definition

A multivariate Gaussian distribution of dimension D can be employed for modelling populations of LE vector space representations of covariance matrices. Its probability density function is given as:

$$p(\mathbf{Y}_{\mathrm{LE}}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{\sqrt{(2\pi)^D|\boldsymbol{\Sigma}|}} \exp\left\{-\frac{1}{2}(\mathbf{Y}_{\mathrm{LE}} - \boldsymbol{\mu})^\dagger \boldsymbol{\Sigma}^{-1}(\mathbf{Y}_{\mathrm{LE}} - \boldsymbol{\mu})\right\} \tag{1.48}$$

where $\boldsymbol{\mu}$ stands for the statistical model's mean of size $[D \times 1]$, $\boldsymbol{\Sigma}$ for its covariance matrix of dimension $[D \times D]$ and $|.|$ denotes the determinant.

1.4.2.2 Parameters Estimation

As in the case of the Riemannian Gaussian distribution defined in the previous paragraphs, the parameters of a multivariate Gaussian distribution are estimated by the MLE method. Let $\mathbf{Y}_{\mathrm{LE}} = \{\mathbf{Y}_{\mathrm{LE}_1}, ...\mathbf{Y}_{\mathrm{LE}_N}\}$ be a set of N i.i.d. data drawn from a multivariate Gaussian distribution of dimension D of parameters $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$. The log-likelihood function holds:

$$\begin{aligned}\mathcal{L}(\mathbf{Y}_{\mathrm{LE}}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) &= \log \prod_{n=1}^{N} p(\mathbf{Y}_{\mathrm{LE}_n}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) = \\ &= -\frac{DN}{2}\log(2\pi) - \frac{N}{2}\log|\boldsymbol{\Sigma}| - \frac{1}{2}\sum_{n=1}^{N}(\mathbf{Y}_{\mathrm{LE}_n} - \boldsymbol{\mu})^\dagger \boldsymbol{\Sigma}^{-1}(\mathbf{Y}_{\mathrm{LE}_n} - \boldsymbol{\mu})\end{aligned} \tag{1.49}$$

and the maximum likelihood estimates of multivariate Gaussian model parameters are:

$$\begin{aligned}\widehat{\boldsymbol{\mu}} &= \frac{1}{N}\sum_{n=1}^{N}\mathbf{Y}_{\mathrm{LE}_n}, \\ \widehat{\boldsymbol{\Sigma}} &= \frac{1}{N-1}\sum_{n=1}^{N}(\mathbf{Y}_{\mathrm{LE}_n} - \widehat{\boldsymbol{\mu}})(\mathbf{Y}_{\mathrm{LE}_n} - \widehat{\boldsymbol{\mu}})^\dagger.\end{aligned} \tag{1.50}$$

In this case, the parameter estimates are given in closed form, as opposed to the statistical models defined on the AI metric space where recursive estimation algorithms need to be used for parameter estimation.

1.4.3 Multivariate Gaussian Mixture Model

Equivalently to the case of the Riemannian Gaussian model previously presented, a mixture model of multivariate Gaussian distributions is proposed on the LE metric space as well, for characterising the variability in data sets of LE mapped covariance matrices. The probability density function corresponding to the mixture model of K multivariate Gaussian distributions consists of the weighted sum of the probability densities of the K Gaussian models, as given below:

$$p(\mathbf{Y}_{\mathrm{LE}}|\omega_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)_{k=1,\ldots,K} = \sum_{k=1}^{K} \omega_k p(\mathbf{Y}_{\mathrm{LE}}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k). \tag{1.51}$$

The mixture model is defined by the following parameters: the means $\boldsymbol{\mu}_k$ and the covariance matrices $\boldsymbol{\Sigma}_k$ of the K multivariate Gaussian distributions and the set of mixture weights $\omega_k > 0$, associated to each mixture component, with $\sum_{k=1}^{K} \omega_k = 1$. The model parameters are estimated by the *Expectation-Maximization* (EM) algorithm (e.g. [Blume 2002]). Details on the algorithm in the specific case of a multivariate Gaussian mixture model are given in Appendix D.

The possibility of representing covariance matrices in vector-forms on the LE metric space has an important practical advantage. It facilitates the characterisation of distributions of n-tuples - groups of n neighbouring covariance matrices. To this purpose, the vector-form covariance matrices forming an n-tuple can be concatenated forming, thus, a vector of dimension $n \times D$. The distribution of the n-tuples can thus be characterised either by a LE multivariate Gaussian model or by its corresponding mixture model of extended size $n \times D$. The characterisation of neighbouring groups of covariance matrices is, hence, made possible in a simple and straightforward way by the LE representation of covariance matrices. On the contrary, this objective is definitely more complex and difficult to envision on the AI metric space. In addition to the complexity of developing such a methodology on the AI metric space, increasing the dimension of the data to be characterised will lead to prohibitive computational costs.

1.4.4 Log-Euclidean Models - Random Sampling

Generating a random sample set $\mathbf{Y}_{\mathrm{LE}} = \{\mathbf{Y}_{\mathrm{LE}_1}, ...\mathbf{Y}_{\mathrm{LE}_N}\}$ from a D dimensional LE multivariate Gaussian distribution of mean $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$ can be achieved in a straightforward way. More precisely, every random vector $\mathbf{Y}_{\mathrm{LE}_i}, i = 1, ...N$ in the sample set $\mathbf{Y}_{\mathrm{LE}}$ is generated by following a series of simple steps [Gentle 2009]:

- find a real matrix $\mathbf{A}$ so that $\mathbf{A}^{\dagger}\mathbf{A} = \boldsymbol{\Sigma}$. This is generally achieved by means of Cholesky decomposition;
- generate a vector $\boldsymbol{x}$ of size $[D \times 1]$, whose elements are independent samples drawn from the standard normal distribution;
- compute the random vector $\mathbf{Y}_{\mathrm{LE}_i}$ following the multivariate Gaussian distribution of parameters $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$:

$$\mathbf{Y}_{\mathrm{LE}_i} = \boldsymbol{\mu} + \mathbf{A}\boldsymbol{x}. \tag{1.52}$$

As in the case of the affine-invariant models, in order to generate a data sample following a mixture model of K multivariate Gaussian distributions, one has to generate samples from each of the K distributions, individually, following the steps previously described. The number of samples

generated from each individual model in the mixture model is proportional to the mixture weights $\omega_k, k = 1, ...K$. The data samples generated from the individual distributions are further concatenated in order to form the randomly generated data sample distributed according to the mixture model of the respective K multivariate Gaussian distributions.

Similar to other operations such as the computation of the center of mass of a sample set of covariance matrices, the task of drawing randomly distributed data samples of a given statistical model is significantly less complex and faster on the log-Euclidean metric space than on the affine-invariant one.

1.4.5 Dissimilarity Measures between LE Models

Jeffrey divergence is employed as dissimilarity measure between two LE statistical models characterising two vector-form samples of LE mapped covariance matrices. Specific details on the evaluation of the Jeffrey divergence for the multivariate Gaussian and the multivariate Gaussian mixture model, are given in the following.

1.4.5.1 Multivariate Gaussian Distributions

In the case of data samples generated from two multivariate Gaussian distribution p and q of unknown parameters, one way to compare them is through a dissimilarity measure between the probability distributions $\hat{p}$ and $\hat{q}$ estimated from those samples. Considering that the parameter estimates of the two probability distributions $\hat{p}$ and $\hat{q}$ are $\widehat{\boldsymbol{\mu}}_1$, $\widehat{\boldsymbol{\Sigma}}_1$ and $\widehat{\boldsymbol{\mu}}_2$, $\widehat{\boldsymbol{\Sigma}}_2$ respectively, the Kullback-Leibler divergence is given in closed form, as follows:

$$D_{KL}(\hat{p}, \hat{q}) = \frac{1}{2}\Big[\log \frac{|\widehat{\boldsymbol{\Sigma}}_2|}{|\widehat{\boldsymbol{\Sigma}}_1|} + \mathrm{Tr}\left(\widehat{\boldsymbol{\Sigma}}_2^{-1}\widehat{\boldsymbol{\Sigma}}_1\right) + (\widehat{\boldsymbol{\mu}}_2 - \widehat{\boldsymbol{\mu}}_1)^{\dagger}\, \widehat{\boldsymbol{\Sigma}}_2^{-1}\, (\widehat{\boldsymbol{\mu}}_2 - \widehat{\boldsymbol{\mu}}_1) - D\Big], \tag{1.53}$$

where D is the dimension of the multivariate Gaussian distribution. The Jeffrey divergence easily follows from equation (1.53), as $D_J(\hat{p}, \hat{q}) = D_{KL}(\hat{p}, \hat{q}) + D_{KL}(\hat{q}, \hat{p})$.

1.4.5.2 Gaussian Mixture Models

When it comes to the comparison of data samples on the LE space characterised by two mixture models of K multivariate Gaussian distributions, the Kullback-Leibler divergence is no longer available in analytical form. Thus, it is approximated by means of Monte Carlo sampling techniques, as previously explained at the beginning of Section 1.3.5. Details on its evaluation, specific to this case, are given in the following.

Let $\mathbf{Y}_{\mathrm{LE}_1} = \{\mathbf{Y}_{\mathrm{LE}_{11}}, ...\mathbf{Y}_{\mathrm{LE}_{1N}}\}$ and $\mathbf{Y}_{\mathrm{LE}_2} = \{\mathbf{Y}_{\mathrm{LE}_{21}}, ...\mathbf{Y}_{\mathrm{LE}_{2N}}\}$ be two samples generated from 2 mixture models of K multivariate Gaussian distributions: $\hat{p}$ of parameters $\widehat{\omega}_{1k}, \widehat{\boldsymbol{\mu}}_{1k}, \widehat{\boldsymbol{\Sigma}}_{1k}$ and $\hat{q}$ of parameters $\widehat{\omega}_{2k}, \widehat{\boldsymbol{\mu}}_{2k}, \widehat{\boldsymbol{\Sigma}}_{2k}$, for $k = 1...K$. The probability density functions for the data in the sample

set $\mathbf{Y}_{\mathrm{LE}_1}$ are given by:

$$\begin{aligned}\hat{p}(\mathbf{Y}_{\mathrm{LE}_{1n}}) &= \sum_{k=1}^{K} \widehat{\omega}_{1k}\hat{p}_k(\widehat{\boldsymbol{\mu}}_{1k}, \widehat{\boldsymbol{\Sigma}}_{1k}) \\ \hat{q}(\mathbf{Y}_{\mathrm{LE}_{1n}}) &= \sum_{k=1}^{K} \widehat{\omega}_{2k}\hat{q}_k(\widehat{\boldsymbol{\mu}}_{2k}, \widehat{\boldsymbol{\Sigma}}_{2k}),\end{aligned} \tag{1.54}$$

where $n = 1, ...N$, $\hat{p}_k \sim \mathcal{G}(\widehat{\boldsymbol{\mu}}_{1k}, \widehat{\boldsymbol{\Sigma}}_{1k})$ and $\hat{q}_k \sim \mathcal{G}(\widehat{\boldsymbol{\mu}}_{2k}, \widehat{\boldsymbol{\Sigma}}_{2k})$. The same relations hold for the data in the sample set $\mathbf{Y}_{\mathrm{LE}_2}$.

By developing the expression in equation (1.30), the approximation of the Jeffrey divergence between two multivariate Gaussian mixture distributions $\hat{p}$ and $\hat{q}$ is as follows:

$$\begin{aligned}\hat{D}_J(\hat{p}, \hat{q}) = \frac{1}{N}\sum_{n=1}^{N}\Bigg\{&\log\sum_{k=1}^{K}\widehat{\omega}_{1k}\left|\widehat{\boldsymbol{\Sigma}}\right|_{1k}^{-1/2}\exp\left[-\frac{1}{2}(\mathbf{Y}_{\mathrm{LE}_{1n}} - \widehat{\boldsymbol{\mu}}_{1k})^{\dagger}\widehat{\boldsymbol{\Sigma}}_{1k}^{-1}(\mathbf{Y}_{\mathrm{LE}_{1n}} - \widehat{\boldsymbol{\mu}}_{1k})\right] - \\ &- \log\sum_{k=1}^{K}\widehat{\omega}_{2k}\left|\widehat{\boldsymbol{\Sigma}}\right|_{2k}^{-1/2}\exp\left[-\frac{1}{2}(\mathbf{Y}_{\mathrm{LE}_{1n}} - \widehat{\boldsymbol{\mu}}_{2k})^{\dagger}\widehat{\boldsymbol{\Sigma}}_{2k}^{-1}(\mathbf{Y}_{\mathrm{LE}_{1n}} - \widehat{\boldsymbol{\mu}}_{2k})\right] + \\ &+ \log\sum_{k=1}^{K}\widehat{\omega}_{2k}\left|\widehat{\boldsymbol{\Sigma}}\right|_{2k}^{-1/2}\exp\left[-\frac{1}{2}(\mathbf{Y}_{\mathrm{LE}_{2n}} - \widehat{\boldsymbol{\mu}}_{2k})^{\dagger}\widehat{\boldsymbol{\Sigma}}_{2k}^{-1}(\mathbf{Y}_{\mathrm{LE}_{2n}} - \widehat{\boldsymbol{\mu}}_{2k})\right] - \\ &- \log\sum_{k=1}^{K}\widehat{\omega}_{1k}\left|\widehat{\boldsymbol{\Sigma}}\right|_{1k}^{-1/2}\exp\left[-\frac{1}{2}(\mathbf{Y}_{\mathrm{LE}_{2n}} - \widehat{\boldsymbol{\mu}}_{1k})^{\dagger}\widehat{\boldsymbol{\Sigma}}_{1k}^{-1}(\mathbf{Y}_{\mathrm{LE}_{2n}} - \widehat{\boldsymbol{\mu}}_{1k})\right]\Bigg\}.\end{aligned} \tag{1.55}$$

1.5 AI and LE Models Representations

The descriptive capacities of the Riemannian Gaussian model and of the multivariate Gaussian model on the LE space are assessed in the following by means of visual representations of randomly generated SPD matrix sample sets. The Riemannian Gaussian distribution and the multivariate Gaussian one have a different number of degrees of freedom. As a consequence, they are expected to have different descriptive capacities.

The number of degrees of freedom of a Riemannian Gaussian distribution of parameters $\mathbf{M}$ and σ, on the space $\mathcal{P}_m$ is given as:

$$DF_{G_{AI}} = \frac{m(m+1)}{2} + 1, \tag{1.56}$$

where the first term of the sum corresponds to the number of degrees of freedom of the center of mass and the second one to the dispersion parameter. The number of degrees of freedom of a multivariate Gaussian model on the LE metric space, corresponding to the same dimension of the SPD matrices space, is given by:

$$\begin{aligned}DF_{G_{LE}} =& \frac{m(m+1)}{2} + \frac{m(m+1)}{2}\frac{\frac{m(m+1)}{2}+1}{2} = \\ =& \frac{m(m+1)}{2} + \frac{m(m+1)[m(m+1)+2]}{8},\end{aligned} \tag{1.57}$$

with the first term of the sum corresponding to the number of degrees of freedom induced by the mean value μ and the second one to the number of degrees of freedom induced by the covariance matrix $\mathbf{\Sigma}$ of the statistical model. In the particular case of $m = 2$, $DF_{G_{AI}} = 4$ and $DF_{G_{LE}} = 9$.

1.5.1 AI - Representations of Various Data Sample Sets Distributions of Specific Known Parameters

In the following, some statistical distributions of random SPD matrix samples of size 2×2 are represented and analysed. The sample sets are drawn from a Riemannian Gaussian model, according to the steps described in Section 1.3.4. Several choices are made for the parameters $\mathbf{M}$ and σ of the theoretical distribution, in order to visualise and assess their impact on the empirical data distributions.

The center of mass $\mathbf{M}$ of the Riemannian Gaussian model is composed using the spectral decomposition given in equation (1.12). The identity matrix $\mathbf{Id}$ is chosen for the matrix $\mathbf{R}$. As for the diagonal matrix $\mathbf{D}$, the $\exp(r_1)$ and $\exp(r_2)$ parameters in equation (1.13) correspond to the covariance matrix's eigenvalues λ_1 and λ_2. Each eigenvalue is associated to one of the 2 orthogonal eigenvectors. The largest eigenvalue corresponds to the eigenvector pointing in the direction of highest variability in the data. In the case when one of the two orthogonal directions is dominant, i. e. $\lambda_1 >> \lambda_2$, the covariance matrix is said to be anisotropic. On the contrary, when there is no privileged direction, in other words, when the eigenvalues are equal or of similar values, the covariance matrix is said to be isotropic. A typical way to asses and to quantify the matrix anisotropy, is by means of the coherence parameter, η, given by:

$$\eta = \frac{\lambda_1 - \lambda_2}{\lambda_1 + \lambda_2}. \tag{1.58}$$

Its value ranges between 0 and 1 and it indicates the degree of confidence that can be assigned to the local orientation estimation. Naturally, a dominant local orientation is associated with a high value of this indicator. More details about the eigenvalue decomposition and about the different ways of exploiting the information encompassed by a covariance matrix in the applied context of texture analysis, are given in Section 2.2.2 of Chapter 2.

Starting from the eigenvalue decomposition, a standard way to visually represent a covariance matrix is by means of an ellipse [Ennis 2006]. The ellipse has its two axes aligned with the covariance matrix's eigenvectors and scaled according to its eigenvalues. Typically, the lengths of its two axes are defined as the square root of the eigenvalues [Ennis 2006, Dryden 2009, Akl 2016].

For randomly generating SPD matrix sample sets following a Riemannian Gaussian model of specific known parameters, the values of the center of mass $\mathbf{M}$ and of the dispersion σ of the theoretical model have to be chosen. Several different scenarios are analysed. As far as the center of mass is concerned, two cases are considered. First, an isotropic choice is made. The eigenvalues are fixed as follows: $\lambda_1 = \lambda_2 = 5$. The identity matrix $\mathbf{Id}$ is chosen for the matrix $\mathbf{R}$. The covariance

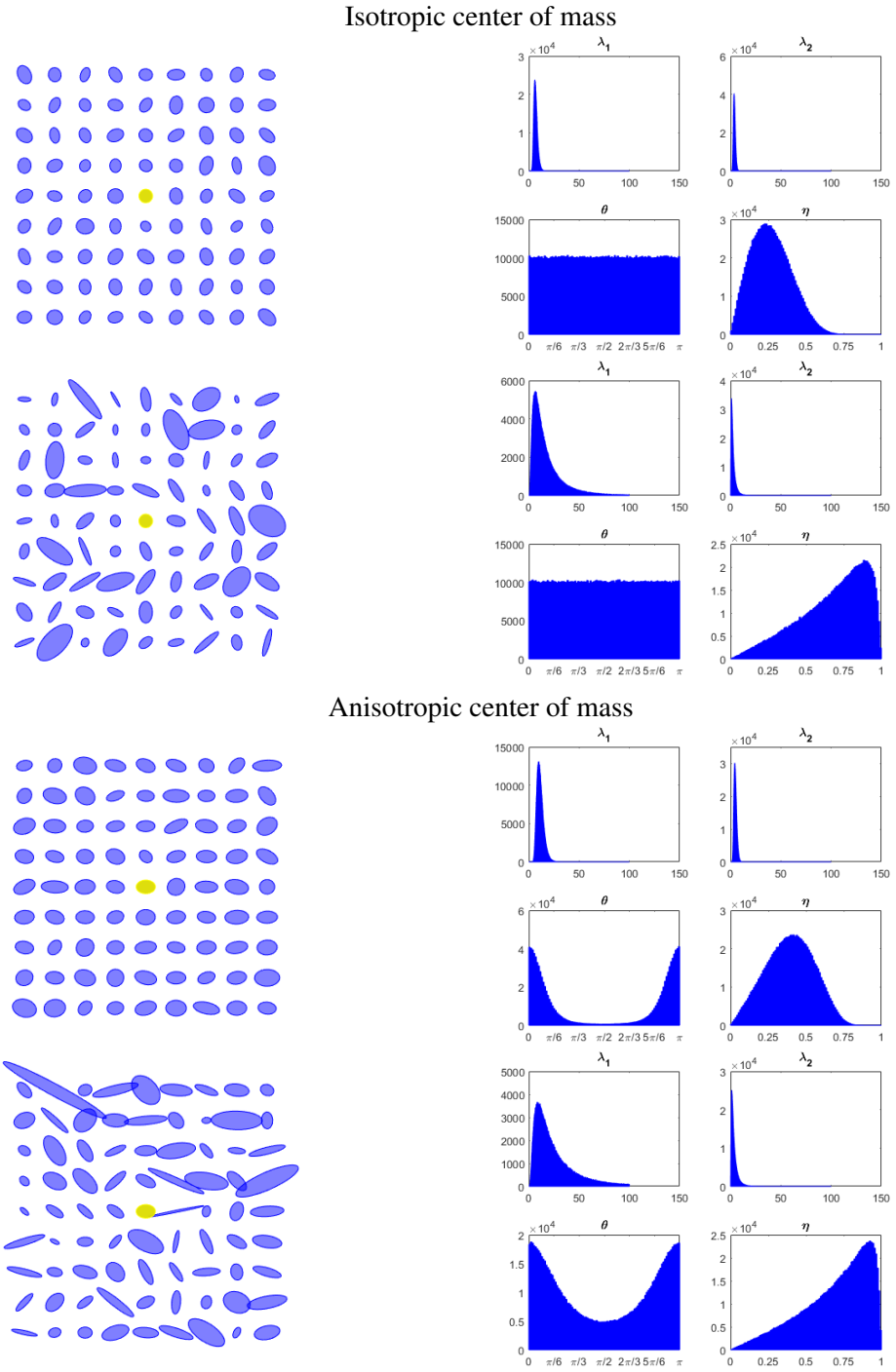


Figure 1.2: Representation of randomly generated SPD matrix sample sets following a Riemannian Gaussian model. Left column: ellipse-form representation of 100 samples. The theoretical distribution's central value is represented in yellow and its variance is assigned a small (first row) and a large value (second row), respectively. Right column: the distributions of parameters of the randomly generated sample set of size 1 million.

matrix corresponding to the center of mass $\mathbf{M}$ of a Riemannian Gaussian distribution is composed, according to equation (1.13). Second, an anisotropic center of mass is considered, with $\lambda_1 = 10$, $\lambda_2 = 5$ and $\mathbf{R} = \mathbf{Id}$, composed according to the steps previously described. For both cases, two values are considered for the theoretical distribution's variance, σ^2: a small value, i.e. 0.1 and a

Table 1.2: Random sampling scenarios - choice of parameters for generating SPD matrix samples following a Riemannian Gaussian distribution of center of mass **M** and dispersion σ

Parametrisation of the center of mass M	**Value of the dispersion σ**
Isotropic: $\lambda_1 = \lambda_2 = 5, \mathbf{R} = \mathbf{Id}$	$\sqrt{0.1}$
	$\sqrt{1}$
Anisotropic: $\lambda_1 = 10, \lambda_2 = 5, \mathbf{R} = \mathbf{Id}$	$\sqrt{0.1}$
	$\sqrt{1}$

large value, i.e. 1. The dispersion σ of the Riemannian Gaussian distribution is computed as the square root of the considered variance. A synthesis of the parametrisation scenarios considered for generating samples from a a Riemannian Gaussian distribution is made in Table 1.2. A sample set of size 1 million is generated for each combination of the theoretical distributions' chosen parameters (see results in Figure 1.2).

Figure 1.2 illustrates, in ellipse-form, subsets of 100 SPD matrices from the randomly generated sample sets of the various Riemannian Gaussian distributions of known parameters. The distributions of the SPD matrices parameters (λ_1, λ_2, θ and η) for the complete randomly generated sample sets are represented as well.

In the case of a small value of the variance, fixed at 0.1, the random SPD matrices in the sample set are naturally more similar to the center of mass of the theoretical distribution, whether it is isotropic or anisotropic. This aspect is visible in the ellipse form representations and equally on the empirical distributions of the parameters of the SPD matrices in the sample set. We can observe that the empirical distributions of the eigenvalues λ_1 and λ_2 are more compact than in the case of a large value of the variance (equal to 1). In addition, the empirical distribution of the coherence parameter, η, has its peak closer to the value that this parameter holds for the center of mass of the theoretical distribution ($\eta = 0$ in the isotropic case and $\eta = 0.33$ in the anisotropic case). The angle θ is differently distributed for the two configurations of the theoretical distribution's central values: uniform distribution when the center of mass is isotropic and compact empirical distribution in the case when the center of mass is anisotropic, having, thus, a privileged orientation.

1.5.2 LE - Representation of Various Data Sample Sets Distributions of Specific Known Parameters

When it comes to the multivariate Gaussian model on the LE space, when fixing the mean μ and the covariance matrix $\mathbf{\Sigma}$ of the theoretical model, the same starting values for the parameters are considered, as in the case of the Riemannian Gaussian distribution. Namely, $\lambda_1 = \lambda_2 = 5, \mathbf{R} = \mathbf{Id}$ for the isotropic case and $\lambda_1 = 10, \lambda_2 = 5, \mathbf{R} = \mathbf{Id}$ for the anisotropic case of the distribution's mean value μ. The center of mass corresponding to these parametrisations is composed in the same way as in the previous case of the Riemannian Gaussian model. An additional step is performed,

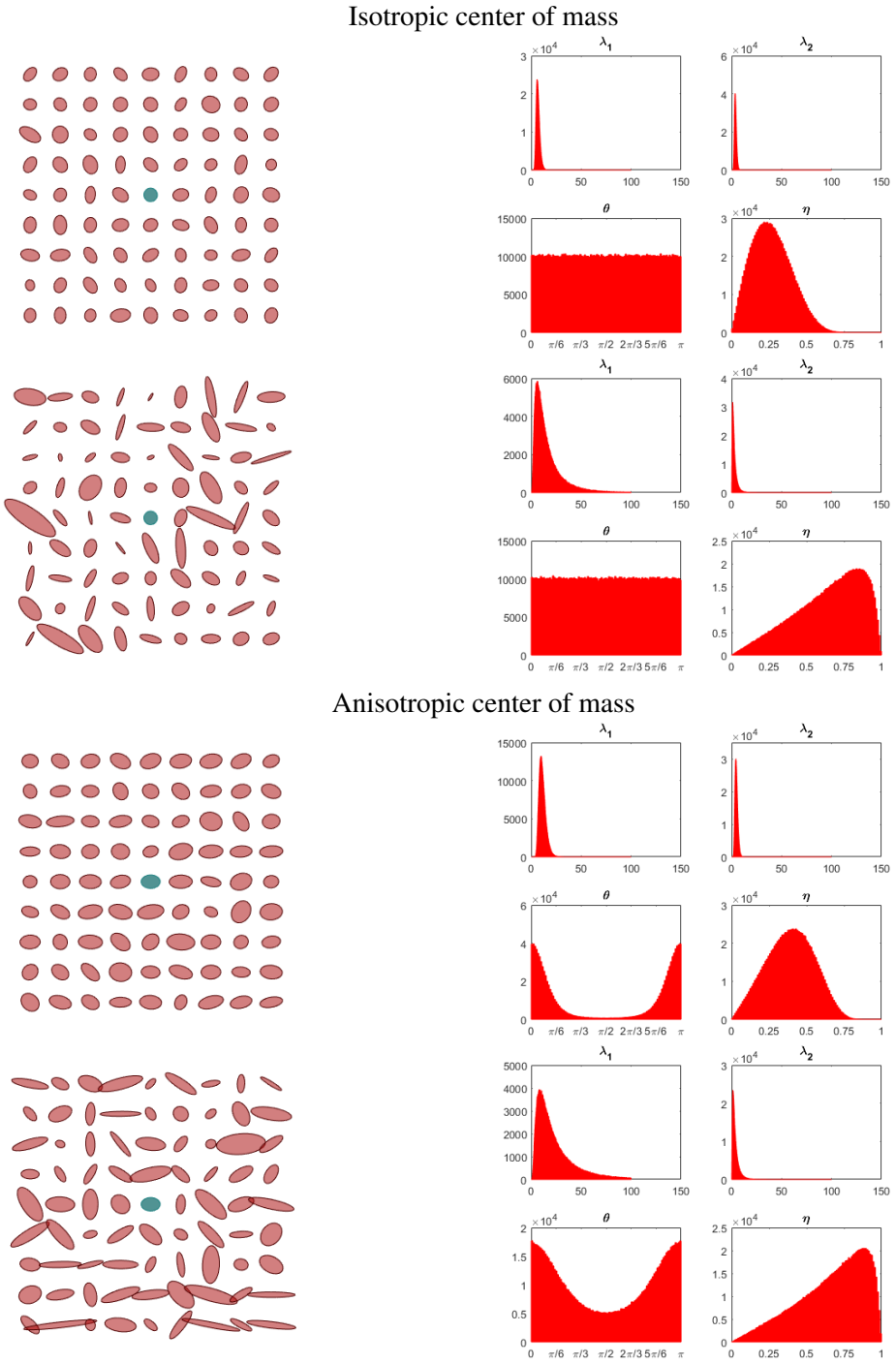


Figure 1.3: Representation of randomly generated SPD matrix sample sets following a LE multivariate Gaussian model of isotropic covariance matrix. Left column: ellipse-form representation of 100 samples. The theoretical distribution's central value is represented in green and its variance is assigned a small (first row) and a large value (second row), respectively. Right column: the distributions of parameters of the randomly generated sample set of size 1 million.

namely, the resulting central element is mapped on the LE space and transformed to its vector-form representation. Thus, the two values considered for the mean $\boldsymbol{\mu}$ correspond to the LE vector-form representations of the values of the center of mass $\mathbf{M}$ previously defined for the Riemannian Gaussian model.

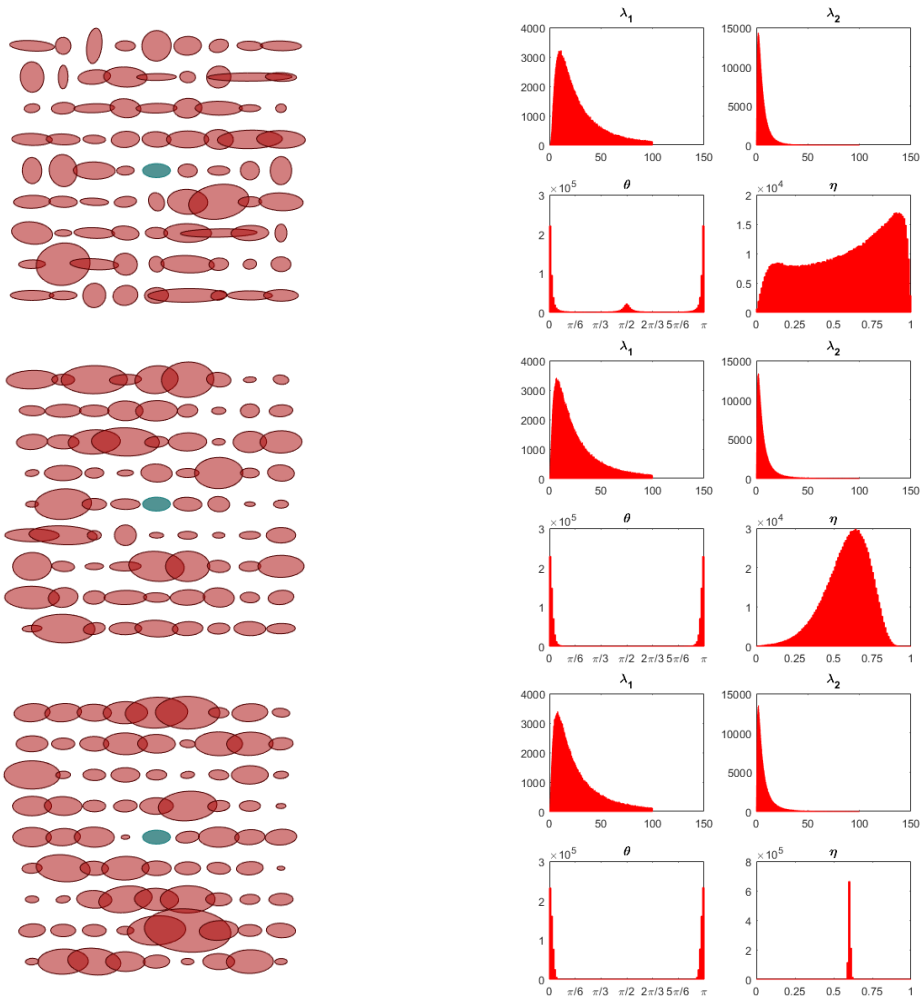


Figure 1.4: Representation of randomly generated SPD matrix sample sets following a LE multivariate Gaussian model of anisotropic covariance matrix. Left column: ellipse-form representation of 100 SPD matrices. The theoretical distribution's central value is represented in green. Right column: the distributions of parameters of the randomly generated sample set of size 1 million. The empirical distributions correspond to a large value for the theoretical distribution's variance.

For both cases, the same two values for the variance σ^2 as in the case of the Riemannian Gaussian model are considered as starting point for computing the theoretical distribution's covariance matrix, namely 0.1 and 1. First of all, a diagonal and isotropic covariance matrix Σ has been composed, of equal variances in all dimensions (see the first part of Table 1.3). The distributions of the random samples following the aforementioned parametrisations are displayed in Figure 1.3.

Second, anisotropic covariance matrix configurations have been considered, with different variance values on the three dimensions and positive covariance between the x and y dimensions of the LE mapped vector form representation of an SPD matrix (see equation (1.44)). The following configurations are considered for the covariance matrix of the LE multivariate Gaussian model (see last part of Table 1.3): diagonal anisotropic covariance matrix (see results on the first row of Figure 1.4), anisotropic covariance matrix, moderately-strong covariance entries between the x and y dimensions (see results on the second row of Figure 1.4), anisotropic covariance matrix, strong covariance en-

Table 1.3: Random sampling scenarios - choice of parameters for generating SPD matrix samples following a multivariate Gaussian distribution of mean $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$

Parametrisation of the mean $\boldsymbol{\mu}$	**Configuration of the covariance matrix $\boldsymbol{\Sigma}$**
	Isotropic
Isotropic: $\lambda_1 = \lambda_2 = 5, \mathbf{R} = \mathbf{Id}$	$\Sigma = \begin{pmatrix} 0.1 & 0 & 0 \\ 0 & 0.1 & 0 \\ 0 & 0 & 0.1 \end{pmatrix}$
	$\Sigma = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$
Anisotropic: $\lambda_1 = 10, \lambda_2 = 5, \mathbf{R} = \mathbf{Id}$	$\Sigma = \begin{pmatrix} 0.1 & 0 & 0 \\ 0 & 0.1 & 0 \\ 0 & 0 & 0.1 \end{pmatrix}$
	$\Sigma = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$
	Anisotropic
Anisotropic: $\lambda_1 = 20, \lambda_2 = 5, \mathbf{R} = \mathbf{Id}$	$\Sigma = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0.01 & 0 \\ 0 & 0 & 1 \end{pmatrix}$
	$\Sigma = \begin{pmatrix} 1 & 0 & 0.9 \\ 0 & 0.01 & 0 \\ 0.9 & 0 & 1 \end{pmatrix}$
	$\Sigma = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 0.01 & 0 \\ 1 & 0 & 1 \end{pmatrix}$

tries between the x and y dimensions (see results on the third row of Figure 1.4). For these scenarios, an anisotropic choice is made for the mean $\boldsymbol{\mu}$ of the theoretical model. It is composed starting from the following parameters' values: $\lambda_1 = 20, \lambda_2 = 5$ and $\theta = 0$. Its associated coherence indicator is equal to 0.6.

As a remark, the sample set is generated on the LE space but its representation is done in the original SPD matrices space, by performing the matrix exponential of the LE matrix-form data. For each chosen parametrisation of the LE multivariate Gaussian model, a sample set of size 1 million has been generated.

The same remarks hold for the randomly generated SPD matrix sample sets drawn from a LE multivariate Gaussian distribution (see results in Figure 1.3), as in the case of a Riemannian Gaussian model (see Section 1.5.1, last paragraph).

1.5.3 AI and LE Models Representations - Final Remarks

While the sample sets illustrated in Figures 1.2 and 1.3 are drawn from two different statistical models, that are, in addition, defined on two different metric spaces, i.e. AI and LE, the empirical distributions of SPD matrices as well as its parameters distributions are very similar, for the particular chosen values and configurations of the parameters of the theoretical distributions.

For a finer analysis of these distributions, the empirical cumulative distribution functions of the coherence parameter η, are illustrated in Figure 1.5, for the AI and LE randomly generated SPD matrix sample sets. For a small values of the theoretical distribution's variance (equal to 0.1), the plots corresponding to the AI and LE are almost identical. However for a higher value of the theoretical variance (equal to 1), there is a more marked difference in the empirical cumulative probability densities of the η parameter of the AI and LE drawn samples.

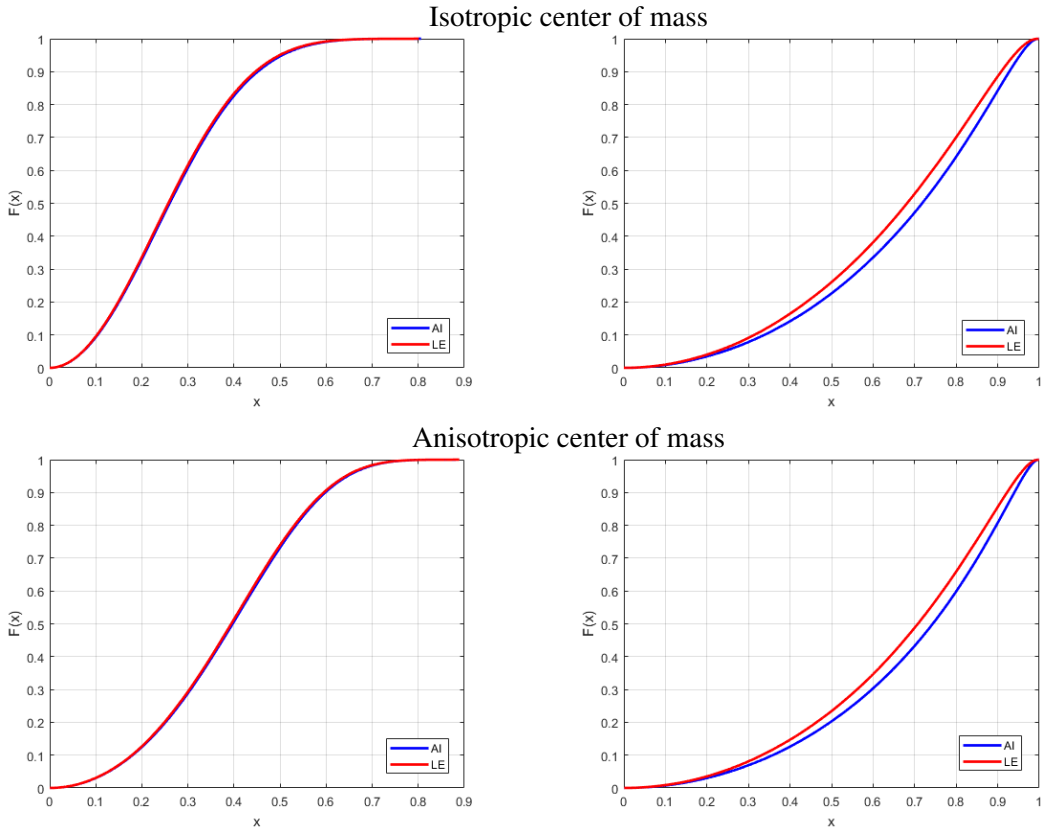


Figure 1.5: Cumulative distribution functions of the coherence parameter of a sample size of 1 million randomly generated SPD matrices generated on the AI (in blue) and LE (in red) metric spaces. The plots correspond to a small (left side) and a large value (right side), respectively, of the theoretical distributions' variances.

In spite of these similarities for these specific scenarios, the LE multivariate Gaussian model has a higher number of degrees of freedom compared to the Riemannian Gaussian model. As a consequence, the choices for the values and configurations of the statistical model's parameters are more abundant. This implies a higher descriptive potential for the multivariate Gaussian model on the LE space as compared to the Riemannian Gaussian distribution defined on the AI metric space. Some examples of SPD sample sets generated upon certain parameter configurations of the multivariate Gaussian model on the LE space, that go beyond the parametrisation capacities of the Riemannian Gaussian model are illustrated in Figure 1.2. An empirical SPD matrix sample set with a very compact distribution of the coherence parameter, i.e. all SPD matrices in the data samples have the same value of the coherence parameter or a very similar one, as the coherence corresponding to the center of mass, is shown in the third row of Figure 1.4. For a characterisation of empirical sample sets as the examples generated in Figure 1.4 - a Riemannian Gaussian distribution is not enough on its own and a mixture model would be necessary for a complete characterisation. In contrast, a single multivariate Gaussian distribution is enough for a complete characterisation of such SPD matrix sample sets. The representations in Figure 1.4 are given as simple examples for illustrating a part of the descriptive potential of a multivariate Gaussian model. However, the descriptive potential

of a multivariate Gaussian distribution is higher.

1.6 Conclusions

In this chapter the space of symmetric positive semi-definite matrices has been introduced. Since Euclidean computations have proven to be inaccurate, non-Euclidean alternatives have been considered for their characterisation. Out of the existing alternatives, the affine-invariant and log-Euclidean metrics were noted as being the strongest in terms of their invariance properties. They are both intrinsic Riemannian metrics defining geodesic distances on the manifold of covariance matrices.

Starting from the proposition of the Riemannian Gaussian model on the affine-invariant metric space and by the idea previously evoked in the literature that the vector-form LE-mapped covariance matrices can be modelled by a multivariate Gaussian distribution, two complete Riemannian frameworks for SPD matrix sample sets characterisation and discrimination have been developed. They rely on the affine-invariant and log-Euclidean metric spaces. In addition to the LE and AI Gaussian models, their corresponding mixture models have been considered as well in order to enrich the capabilities of the theoretical distributions to encompass the variability that arises in real-world data. The parameter estimation process is detailed for every case.

Besides characterising covariance matrix data samples, a real need in some applications is their comparison. This task can be achieved by comparing the estimated probability distributions characterising data samples of SPD matrices. To this purpose, Jeffrey divergence is chosen as dissimilarity measure between probability distributions. Solutions for its computation are given for each statistical model. In the cases when its expression is not available in analytical form, solutions for its approximation based on Monte Carlo sampling techniques are given.

While the AI metric space is endowed with stronger invariance properties, estimating the parameters of the statistical models relies on recursive estimation algorithms, inducing thus high computational expenses. The LE metric is endowed with similar invariance properties as the AI one. However, as the LE mapping allows the vector-form representation of covariance matrices, the complexity and computational expenses associated to the algorithms on the LE metric space are significantly reduced.

The distributions of SPD matrix sample sets following a Riemannian Gaussian model and a LE multivariate Gaussian model of fixed parameters have been represented and analysed, for various variances of the theoretical distributions, in order to illustrate the descriptive potential of the statistical models. Justified by a higher number of degrees of freedom for a same order of the theoretical models, the LE models showed a grater capacity that the AI models to characterise the variability arising in SPD matrix sample sets.

Chapter 2

Describing Structure in Textures through LST Field Modelling

"...texture is what makes life beautiful; texture is what makes life interesting and texture is what makes life possible. Texture is what makes Mozart's music beautiful, the masterpieces of the art of the Renaissance classical and the facades of Barcelona's buildings attractive. Variety in detail is what keeps us going from one day to the next and the roughness of the world is what allows us to walk, communicate and exist. If surfaces were smooth, friction would not exist, the Earth would be bombarded with asteroids and life would not have developed. If surfaces were smooth, pencils would not write, cars would not run, and feet would not keep us upright."

— Maria Petrou and Pedro García Sevilla, authors of *Image Processing: Dealing with Texture* [Petrou 2006]

Contents

2.1 Texture

2.1.1 What is a Texture?

There is no universally agreed upon answer to this question. While recognizing texture is an innate and automatic task for the human visual perception, defining texture comes less naturally. Its definition varies from the most simple and general ones *"the feel, appearance, or consistency of a surface or a substance"* [Oxf 1992], to the most intricate and even philosophical ones, as the definition in the epigraph of this chapter. Therefore, the notion of texture is quite complex.

An early scientific definition of textures from the perpective of their visual interpretation and discrimination is given by Julesz et al. [Julesz 1983]. Textures are viewed as *"aggregates of many small elements. The elements can be either dots of certain colors (e.g. white, black, gray, red) or simple patterns"*. Haralick [Haralick 1979] views a texture as a function of its composing tonal primitives. Namely, a texture is defined by the types and number of primitives along with their spatial organisation. Textures can generally be seen in the field of computer vision as functions of spatial variations in the pixels' intensity values [Conners 1980] or as *"images containing repeated patterns"* [Wei 2009], allowing for a certain degree of randomness over these patterns.

2.1.2 Texture in Visual Perception

Texture is at the core of human visual perception. While it comes naturally to the human eye to identify and discriminate textures, in digital applications these tasks need to be learned. Texture description and classification are strictly dependent on identifying and extracting those features that make a texture unique or particular. Understanding the processes involved in the visual perception of a textured scene and identifying those key texture characteristics that are fundamental to the human eye when discriminating textures, is a stepping stone for developing algorithms in computer vision applications.

Pioneering research has been conducted on texture by Julesz et al. [Julesz 1962, Julesz 1978, Julesz 1983], in an attempt to develop a model for the human visual perception mechanism by assessing the psycho-physiological processes involved when interpreting and discriminating texture. A first study [Julesz 1962] revealed that textures differing in their first and second order statistics are interpreted as being distinct by the human visual perception. This theory was contradicted by Julesz himself in a later study [Julesz 1978] that showed classes of textures that, while having identical third order statistics, are easily discriminated by the human brain. In a later study the textons theory has been developed [Julesz 1983], where a texture is seen as a function of image primitives called textons. This research shows that the pre-attentive vision discriminates textures when there is a difference in the densities of textons. The importance of texture in the human visual perception is reinforced by a different study [Haralick 1973], identifying texture as one of the 3-key elements that humans use when identifying the regions of a gray-level image, in addition to tone and context.

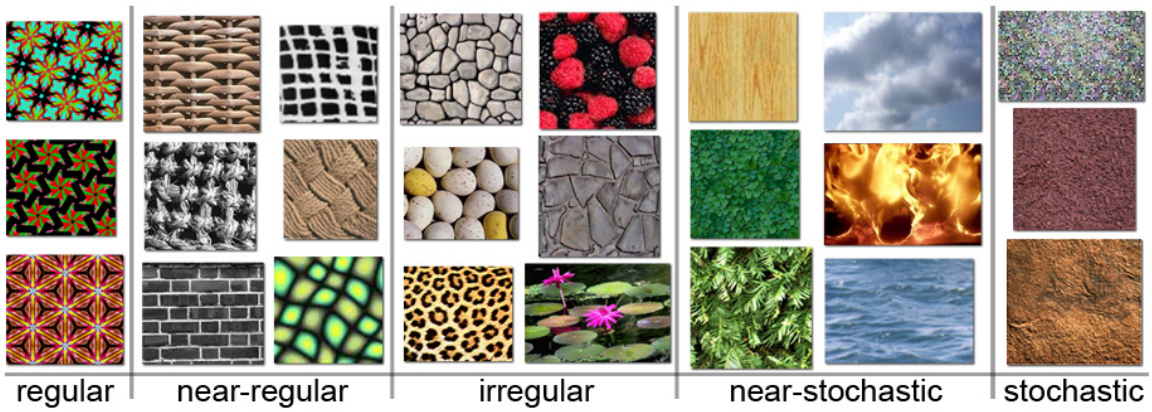


Figure 2.1: Texture spectrum in terms of regularity [Hayes 2006]

Inspired by these findings in the field of visual perception, texture becomes a key-element when processing a digital image. Different databases comprising a large variety of either color (VisTex [MIT 1995], Outex [Ojala 2002a]) or gray-level (Brodatz [Brodatz 1966]) textures are publicly made available, for testing and benchmarking new texture analysis algorithms.

2.1.3 Types and Properties

Many attempts have been made in the literature to identify a set of properties that are sufficient for characterising the variety of a texture's content.

A core property of textures is the homogeneity [Unser 1984]. In terms of visual perception, homogeneity can be explained as follows: given an analysis window of fixed size, if one slides the window over the texture, the visual content through the observation window has the same characteristics, no matter the location of the analysis window over the texture [Unser 1984]. In statistics term, homogeneity translates as stationarity. If the texture is viewed as the realisation of a random process, the stationarity refers to the translation invariance of the statistical properties defining the random process. The same view of a texture in terms of constant or slowly varying local statistic has previously been suggested by Sklansky in [Sklansky 1977].

Tamura et al. [Tamura 1978] proposed a set of 6 fundamental properties considered to be characteristic to all textures. A first property is coarseness. It is seen as a central characteristic of textures by several authors [Hayes 1974, Amadasun 1989]. For coarse textures, the composing primitives or repeating patterns have a large size, as opposed to fine textures characterised by a small size of their composing patterns. Contrast is considered to be another basic property shared by all textures, accounting for the amount of change in the texture's intensity values. Two other properties are proposed for describing a texture's orientation: directionality used as a global indicator and line-likeliness used to locally describe the shape of the texture's primitives (line-like vs. blob-like). A texture can be characterised in terms of its regularity [Efros 1999], on a spectrum going from highly-structured (regular) to random (irregular) textures, as illustrated in Figure 2.1. Artificially

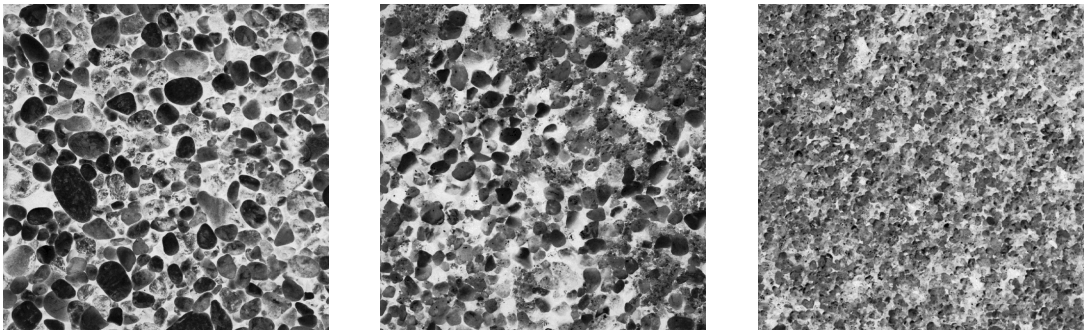


Figure 2.2: Examples of pebbles textures from the Brodatz database going from macro-textures to micro-textures (from left to right).

generated textures can be regular, namely, composed of a perfect repetition of the same unique pattern. However, the majority of natural textures are situated somewhere in between near-regular and stochastic textures.

Another set of basic properties common to most textures is proposed in [Amadasun 1989]. Three new texture properties are introduced. They consist of busyness (quantifying the amount of spatial changes in the intensity levels), complexity (assessing the information diversity of the texture's content) and texture strength (measuring the perception intensity of the basic repeating elements, in terms of clarity and visibility).

Some of the properties proposed by Tamura [Tamura 1978] and Amadasun [Amadasun 1989] have turned out to be strongly correlated. In an attempt to find the minimum set of texture characteristics that are enough for describing and discriminating textures, Rao et el. [Rao 1993] conducted an analysis of the feature space of textures. Their experiments concluded that only three of the investigated features are responsible for most of the data variability (over 96%). These features are orientation (that can be interpreted as the combination of directionality and line-likeliness features proposed in [Tamura 1978], that, in reality, are strongly correlated), repetition (that can be viewed as the degree of regularity) and complexity. An interesting analogy is made between color that on the RGB (Red, Green, Blue) color space is defined by its 3 primary colors and texture that can be characterised in terms of its 3 primary properties, namely orientation, repetition and complexity [Rao 1993].

According to the type of the observed objects or to the scale of observation, textures can be divided in two classes [Galerne 2011]. A first class is represented by micro-textures, that can be seen as homogeneous, stochastic textures. Natural micro-textures can be represented by sand, water, etc. The second class is represented by macro-textures, or textures that illustrate small size objects that are clearly visible on the image. Some examples of macro-texture include pebbles, brick walls or tiled roofs. In addition to the nature of the observed objects, the distance from the observed scene dictates the micro or macro character of a texture. For example, a pebbles texture observed from a close distance will be considered a macro-textures, since the individual pebbles will be clearly visible on the image. However, if the observation distance overpasses a certain threshold, the indivi-

dual pebbles will no longer be discernible on the texture. Instead, the texture will have a stochastic aspect, so it can be considered a micro-texture (see examples in Figure 2.2). Therefore, the nature of a texture is deeply related to the scale of observation and analysis. As a consequence, it is important for texture analysis tools used in computer vision to be able to adapt the scale of the analysis window to the scale that is relevant for the textural content that one wants to describe.

2.1.4 Texture Application Fields

Texture appears in a wide range of application fields in computer vision. Developing algorithms for analysing texture allows the automation of many tasks otherwise performed by humans. Thus, once time-consuming tasks are instantly performed with a lower need of resources.

Texture analysis methods are commonly applied for tasks such as segmentation (dividing an image into different regions, according to their similarity), classification (groping the textures in a database in classes according to some pre-defined similarity criteria) or they can consist of a stepping stone for texture synthesis (creating a synthetic texture visually very similar to an input texture, given as example).

The emerging developments in medical imaging technology create the need of innovating algorithms and tools for the analysis of clinical data [Duncan 2000]. Texture analysis techniques are used to help clinicians in the diagnosis process, among other clinical procedures [Castellano 2004]. Texture analysis methods can be employed for identifying and extracting target anatomical objects in medical images. The results of segmentation can further be used by clinicians for extracting different statistics, such as the size and shape of tumours or other abnormal anatomical structures. Texture analysis algorithms can be employed as well for tissue analysis in tasks such as lesion detection or discrimination between normal and pathological tissues.

Texture analysis methods are extensively used in remote sensing for extracting the local geometry information characterising the terrain organization and dynamics [Aptoula 2014, Regniers 2016]. The extracted information can further be used for evaluating meaningful statistics or for assessing and quantifying the impact of human activities and different natural phenomena, such as severe meteorological events. The applications are rich and diverse, including forestry and urban remote sensing data classification [Ruiz 2004], dead vine trees detection in airborne remote sensing vineyard images [Chanussot 2005], delineation of vine parcels in VHR remote sensing images [Da Costa 2007], classification of oyster parks [Regniers 2015b].

Texture analysis methods are used as well in the industrial quality control process for developing automated inspection systems aimed to replace the otherwise human-performed visual inspection, that is quite a laborious task. Instead, defects can be automatically detected by analysing the textured images corresponding to the surfaces of different types of materials, such as: metals, ceramics, textiles (e.g. [Tomczak 2007]).

Texture is omnipresent in the field of computer graphics, where creating a virtual reality visually

similar to the actual scenes of the real-world is an ongoing challenge. Texture synthesis algorithms can be used to this purpose. Different solutions have been proposed in the literature, including [Efros 1999, Wei 2000, Galerne 2011, Urs 2013, Akl 2015, Gatys 2015].

Document processing is another application field for textures. Texture analysis algorithms can be used in a segmentation pre-processing step of document images, for identifying regions of interests. Such regions can be represented by the text blocks in a newspaper, a bar code on a document or the address on an envelope [Jain 1992]. The segmented data can be used in different purposes, such as address recognition or restoration of old text.

2.1.5 Texture Analysis

Given the ubiquitous character of a texture, developing texture analysis methods that are easy to apply, fast and that have a rich descriptive potential is a real need for many applications. A wide range of texture analysis methods are proposed in the literature. Different categorisations of these methods have been considered. In this work, the following one is adopted (inspired by [Materka 1998]):

- structural methods - inpired by the texon theory [Julesz 1983], these approaches describe a texture in terms of its primitives, by analysing their types and the spatial arrangements between these primitives [Haralick 1979];

- descriptive statistics based methods - inspired as well by the research conducted by Julesz [Julesz 1962], these texture analysis methods describe the distribution of a texture's gray levels by means of second order statistics. In other words, these methods rely on the analysis of the spatial organisation of a texture's intensity values. The most popular approach belonging to this category is represented by the gray level co-ocurrence matrix (GLCM) [Haralick 1973]. Some other texture anlysis methods belonging to this group are: local binary patterns (LBP) [Ojala 1994], auto features [Tuceryan 1993] and variograms [Curran 1988];

- frequency domain filtering based methods - are inspired by the research conducted on visual perception, concluding that the visual cortex decomposes an observed image into components tuned to different frequencies and orientations [Conners 1980]. The texture analysis methods from this category are based on spectral decomposition methods, such as the Fourier transform (e.g. [Zhou 2001], the Gabor filter [Turner 1986] and the wavelet transform [Mallat 1989]);

- model based approaches - are describing texture by means of a statistical model, considering texture as the realisation of a stochastic process. Some examples include Markov random field (MRFs) models [Cross 1983], or different statistical models applied on previously extracted texture features. These features can be represented by the wavelet subband coefficients [Do 2002, Regniers 2014b, Regniers 2014c] or by the response of Gabor filtering [Mathiassen 2002]. The univariate statistical models used to describe distributions of

spectral coefficients include the generalized Gaussian distribution [Do 2002], Gamma distributions [Mathiassen 2002], Bessel K forms [Srivastava 2002]. Multivariate models have been employed as well, such as: multivariate Gaussian distribution [Ilea 2017], multivariate generalized Gaussian distribution [Verdoolaege 2011], SIRV (Spherically Invariant Random Vectors) [Regniers 2015a], multivariate Bessel K form distributions [Boubchir 2010], copula based distribution [Kwitt 2009].

We will briefly describe in the following the state of the art texture analysis methods that will be later used in the experimental part of Chapter 3 for benchmarking the approaches proposed in this work. These methods belong to the last 3 categories of texture analysis methods, considering the previously presented categorization.

2.1.5.1 Descriptive Statistics based Methods

Gray Level Co-occurrence Matrix

Proposed by Haralick et al. [Haralick 1973], the gray level co-occurrence matrix (GLCM) is describing a texture in terms of its gray levels spatial organisation. It quantifies the number of occurrences of every pair of gray levels in the texture, according to a specific spatial configuration. The configuration is given by the distance between the pairs of pixels and by a displacement vector of components dx and dy. For an image $\boldsymbol{I}$ of size $w \times h$ with L different intensity values, the co-occurrence matrix of size $L \times L$ is computed as follows:

$$\mathrm{GLCM}_{dx,dy}(i,j) = \sum_{x=1}^{w} \sum_{y=1}^{h} \begin{cases} 1, \text{if } \boldsymbol{I}(x,y) = i \text{ and } I(x+dx, y+dy) = j \\ 0, \text{otherwise,} \end{cases} \tag{2.1}$$

where i, j represent the intensity values of the current pixel and of its neighbour pair, respectively.

In general, the GLCM result as given by equation (2.1) is not directly employed. Instead, the GLCM content is summarised by computing different statistical descriptors [Haralick 1973, Conners 1984]. Among the proposed GLCM descriptors, Maillard et el. [Maillard 2003] have identified a set of 5 that are most widely used in the literature. This set consists of the following features: energy, entropy, contrast, correlation and homogeneity.

Local Binary Patterns

Originally proposed by Ojala et al. [Ojala 1994], this approach consist of encoding the spatial dependence of the gray levels in the neighbourhood of a pixel by a binary sequence, hence the name *local binary patterns* (LBP). In its original version, 3×3 neighbourhoods are considered and the binary sequence in each pixel is obtained by tresholding the intensity values in the neighbourhood with the intensity value of the central pixel. If the neighbouring pixel has a greater intensity value than the central one, a value of 1 is assigned and 0 otherwise, resulting thus in an 8 long binary

sequence. The binary sequence of each pixel is assigned a label out of the $2^8 = 256$ different possible values and the histogram of these labels is further computed and considered as texture descriptor.

Different variations from the original descriptor have been proposed, adapted for larger neighbourhoods [Ojala 2002b]. The propositions include multiscale and rotation invariant local binary patterns.

2.1.5.2 Frequency Domain Filtering based Methods

Gabor Filters

Research has shown that the cells of the visual cortex respond to an observed scene by decomposing the visual scene into bands of different frequencies and orientations [Marĉelja 1980]. The perception mechanism can be modelled by Gabor filters [Marĉelja 1980], that perform a multi-resolution and multi-orientation analysis, when applied on a texture. In the spatial domain, Gabor filters are defined as Gaussian functions modulated by a sinusoid [Jain 1990]. In the spectral domain, a Gabor filter is a Gaussian function centered on a frequency f and an orientation θ.

In general, Gabor filter banks are applied for characterising a texture, where each filter is centered on a given frequency and orientation in an attempt to obtain a complete coverage of the spectral domain [Manjunath 1996]. The response of the filter banks in each pixel of the image are further combined for obtaining the textural signature. In spite of the capacity of Gabor filters to perform a multi-resolution and multi-orientation analysis of a texture, the filter response might be correlated, given the non-orthogonality of the filter-banks.

Orthogonal Wavelet Transform

The orthogonal wavalet transform is another way of performing a multi-resolution analysis of a texture [Mallat 1989]. By employing a series of discrete wavelet functions, the texture is decomposed into independent and orthogonal subbands. Some of the limitations of Gabor filtering, such as the correlation of the extracted textural signatures, are overcome as a result.

The multiscale wavelet analysis of an image is achieved by decomposing the image, at each scale of analysis, into 4 subbands down-sampled by a factor of 2. The decomposition is performed by applying separable low-pass (L) and high-pass filter banks (H) along the rows and columns of the image. The 4 subbands of wavelet coefficients LL, LH, HL, HH are obtained as a result of the combination of the low-pass and high-pass filters. LL represents an approximation subband as opposed to the other three that contain the high-frequency details. In consequence, uniquely the coefficients of the LH, HL, HH subbands are exploited for computing the textural signature. In the following, we simply employ the terminology *wavelet transform* when referring to the orthogonal wavelet transforms.

2.1.5.3 Model based Methods

Among the model based approaches, the ones that have been chosen for comparison in this work are statistical models applied for characterising a texture's wavelet subband coefficients [Bombrun 2011b, Lasmar 2014, Regniers 2014a, Regniers 2014c].

Multivariate Gaussian Model

A first statistical model that has been used for describing the distribution of a texture's wavelet subband coefficients is the multivariate Gaussian model, e.g. [Regniers 2014a].

Spherically Invariant Random Vectors

Introduced by Yao [Yao 1973], spherically invariant random vectors (SIRV) models are a family of Gaussian non-homogeneous distributions of random variances. An observed vector $\boldsymbol{v}$ is expressed as:

$$\boldsymbol{v} = \sqrt{\tau}\boldsymbol{z} \tag{2.2}$$

where $\tau > 0$ is a random variable and $\boldsymbol{z}$ is a Gaussian random vector of mean 0 and covariance matrix $\boldsymbol{\Sigma}$. The variable τ does not follow a pre-defined statistical model. Consequently, upon the choice of the statistical model for the variable τ, the SIRV model will have different definitions [Bombrun 2011a]. A SIRV Gaussian model results in the case when the τ parameter follows a Dirac distribution.

Copula based Distributions

Several configurations are possible for these statistical models. The one used for comparison in this work proposed by [Stitou 2009] and applied by Regniers in [Regniers 2014a] is a Gaussian copula with the marginal distribution following a Gamma model.

2.2 LST - Local Structure Tensor for Texture Analysis

The local structure tensor [Bigün 1987, Knutsson 1989], known in the literature as the second moment matrix as well, is a commonly employed tool in computer vision for describing the structure of an image. It is often employed in textured image analysis for extracting the geometric information that characterises locally a texture's patterns. The characterisation of the local pattern is achived by estimating the main directional tendencies in the neighbourhood of a point [Jähne 1993]. The coherence of the gradients orientations around the estimated main directional tendencies is evaluated as well.

A common structure tensor application consists of edge and corner detection [Harris 1988, Arseneau 2006]. Another application example is fault detection in seismic data [Donias 2007]. In this case, the structure tensor is exploited for measuring the disorder of the gradient field, a

high value of this measure being associated with the presence of a fault. An LST field regularisation method based on bilateral filtering is proposed by Toujas et al. [Toujas 2010] in the aim of analysing the structure of seismic images. The LST is also employed for texture segmentation [Rousson 2003, Pham 2015b]. The second study proposes a covariance matrix method integrating the intensity levels and the oriented gradients as well. Furthermore, the local structure gradient is used in addition to polarimetric information for classifying synthetic aperture radar (SAR) images [Pham 2015a]. In addition to characterisation and identification tasks, the structure tensor method is used as well for more complex tasks such as texture synthesis [Peyré 2010, Akl 2014, Akl 2015].

2.2.1 LST Computation

There are two families of methods that can be employed for structure tensors estimation [Aja-Fernández 2009]. They consist of gradient methods [Bigün 1987, Kass 1987, Förstner 1987] and local-energy methods where quadrature filters are used for locally characterising the image structure [Knutsson 1989]. In this work the structure tensor is estimated by a gradient based approach.

Let I be an image of N pixels. The gradient of the image ∇I is firstly computed by convolving the image with spatial derivative filters:

$$\nabla I = \left[I_x, I_y\right]^\dagger = \left[I * G_x, I * G_y\right]^\dagger , \tag{2.3}$$

where I_x and I_y are the horizontal and vertical partial derivative estimates of the image I and $*$ stands for the convolution operator. G_x and G_y are two filters used for estimating the image gradient in the horizontal and vertical direction, respectively. They can take the form of basic discrete gradient estimation filters, such as the Robert [Dav 1975] or Sobel operators [Lyvers 1988] that are fast and easy to implement. In this work the gradient is estimated by linear convolution with Gaussian spatial derivative filters given as the first order partial derivatives of a 2 dimensional Gaussian [Perona 1990, Rao 1991]:

$$G_x(x,y) = -\frac{x}{2\pi\sigma_G^4}\exp\left\{-\frac{x^2+y^2}{2\sigma_G^2}\right\}, G_y(x,y) = -\frac{y}{2\pi\sigma_G^4}\exp\left\{-\frac{x^2+y^2}{2\sigma_G^2}\right\}. \tag{2.4}$$

These filters allow to adjust the size of the local neighbourhood considered for gradient estimation, by tuning the value of the standard deviation σ_G, as opposed to the basic derivative operators that are of fixed size.

In order to describe the structure at a given pixel in the image, one should analyse the data in the neighbourhood of that point. To this purpose, the outer product of the gradient is convolved with a 2 dimensional Gaussian smoothing kernel W_T, whose standard deviation, σ_T, defines the size of the local neighbourhood. In addition to enabling a local analysis at a tunable scale, the smoothing kernel increases the robustness of the method to noise or other undesirable image artefacts. Hence,

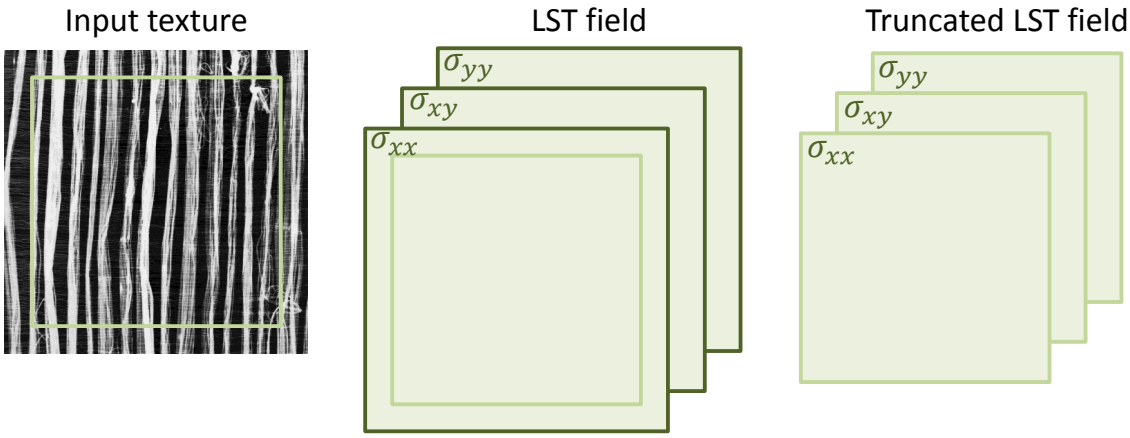


Figure 2.3: LST field border effect. The border induced by successive filtering is marked in green on the input texture. The 3 component LST field before and after truncation are illustrated on the second and third column.

the local structure tensor field $\mathbf{Y}$ is given as:

$$\mathbf{Y} = W_T * \nabla I \nabla I^{\dagger} = W_T * \begin{bmatrix} I_x.I_x & I_x.I_y \\ I_x.I_y & I_y.I_y \end{bmatrix}, \tag{2.5}$$

where . denotes point-wise multiplication and $*$ denotes convolution. The resulting matrix-valued LST field has the same size as the image. Thus, every pixel n in the image, $n = 1...N$, is associated a local structure tensor in the form of a 2×2 symmetric non-negative definite matrix:

$$\mathbf{Y}(n) = \begin{bmatrix} \sigma_{xx}(n) & \sigma_{xy}(n) \\ \sigma_{xy}(n) & \sigma_{yy}(n) \end{bmatrix}. \tag{2.6}$$

However, in practice, the LST field is truncated, as illustrated in Figure 2.3, given that there is a border effect induced by the successive spatial filtering. The size of the image border that needs to be removed depends on the dimension of the derivative and smoothing Gaussian kernels. Typically, the derivative Gaussian kernels G_x and G_y and, respectively, the Gaussian weighting kernel W_T are truncated at $3\sigma_G$ and $3\sigma_T$, respectively, resulting in border effects within margins of size $3\sigma_G + 3\sigma_T$ that should be removed.

2.2.2 LST Decomposition and Representation

The LST field describing a textured image is difficult to represent and visualise in raw form. Simplified representations can be achieved by mapping the tensor values to scalar quantities. A collection of such scalar-valued descriptors, called tensor invariants is given in the context of DT-MRI for the diffusion tensors [Papadakis 1999, Ennis 2006]. In contrast, the structure tensor literature lacks such a study. The structure tensor information is generally quantified in terms of its orientation and eigenvalues that can be further exploited for computing different local descriptors, such as the energy

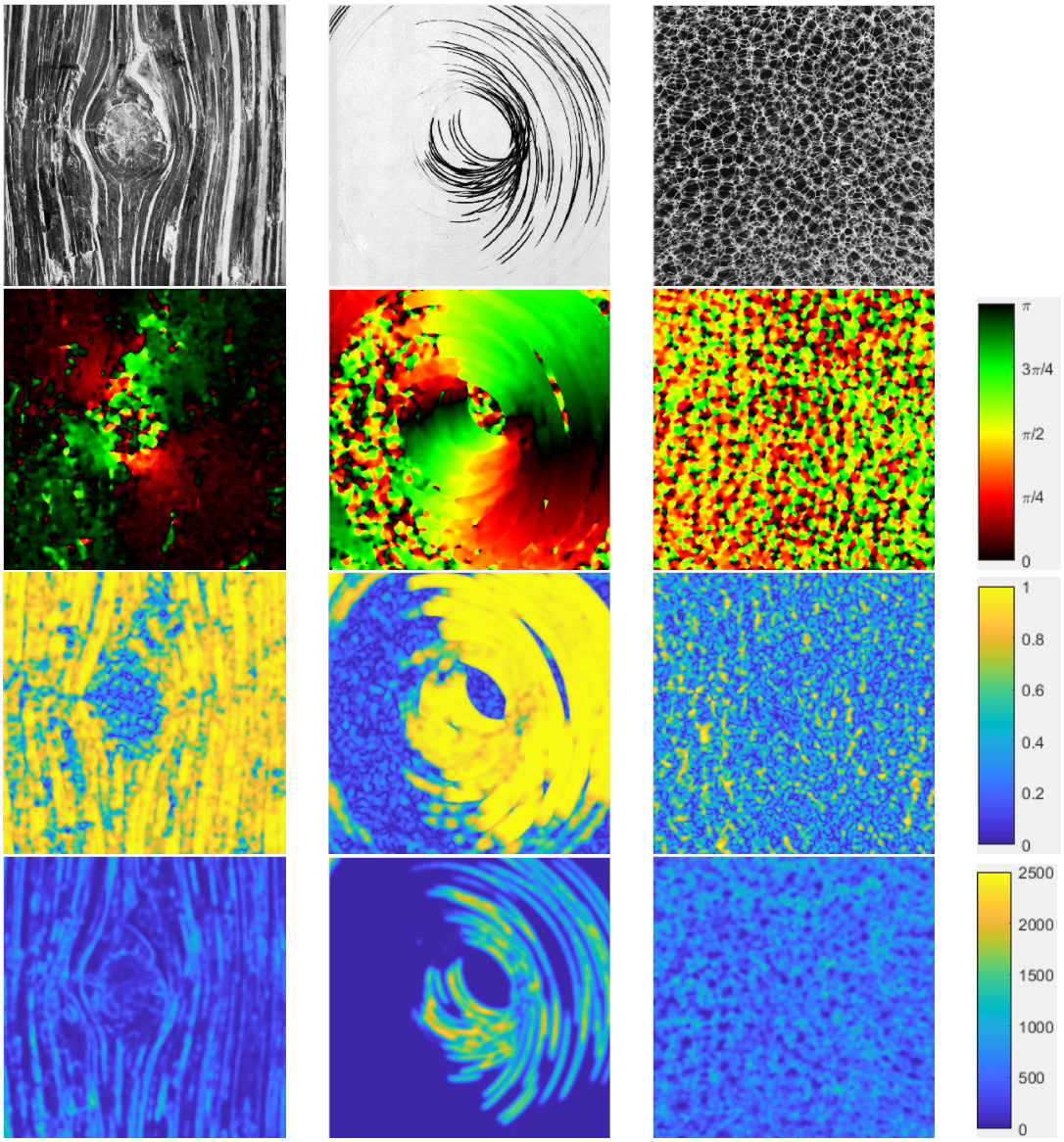


Figure 2.4: Examples of textures from Brodatz database (first row) and their corresponding LST parameters fields (from second to forth row): orientation, coherence and energy. The LST fields were calculated for the following values of the parameters: $\sigma_G = 1$ for all textures and $\sigma_T = 4, 5$ and 4, respectively (considering the textures from left to right).

and the coherence.

The texture geometry information extracted by the structure tensor mainly consists of the orientation estimated in the neighbourhood of a point by symmetric non-negative definite matrix [Knutsson 1989]. However, in addition to the orientation, the structure tensor gives as well a measure of the certainty associated to the estimated local orientation. This property is sometimes referred to as *anisotropy*. In other words, it indicates to which extent the orientation in the neighbourhood of a point is dominant.

These local indicators of a texture's geometry are obtained by eigenvalue decomposition. The eigenvectors at a point n, $\mathbf{u}_1(n) = [u_{1x}(n), u_{1y}(n)]^\dagger$ and $u_2(n) = [u_{2x}(n), u_{2y}(n)]^\dagger$ can be used to

characterize the texture's directional tendency around that point. More precisely, the first eigenvector $\mathbf{u}_1(n)$ indicates the direction of strongest intensity variation and is, thus, orthogonal to the local pattern orientation. It gives an estimation of the gradients' mean orientation $\theta(n)$, in the neighbourhood of a point n. The orientation angle is calculated considering a coordinate system with the x axis pointing to the left and the y axis pointing downwards (typical image coordinate system). The gradient's orientation corresponds to the angle the eigenvector makes with the x axis, measured in the conventional clockwise direction. It takes values in the interval $[0, \pi]$ ($\theta(n) = \theta(n) + \pi$ - the same angle is considered for eigenvectors having the same orientation but opposite senses). The second eigenvector $\mathbf{u}_2(n)$, orthogonal to $\mathbf{u}_1(n)$, points along the texture pattern and it can be used for estimating the texture's orientation, in the neighbourhood of the point n.

In addition to the eigenvectors, their corresponding eigenvalues $\lambda_1(n)$ and $\lambda_2(n)$ can be exploited to compute local anisotropy or energy indicators in the neighbourhood of the point n. The degree of anisotropy of a texture is an indicator of its directional character and can be assessed by evaluating the gap between the two eigenvalues. More precisely, a large gap between their values, $\lambda_1(n) \gg \lambda_2(n)$, indicates a preferred gradient orientation in the neighbourhood of the point n and it is a sign of strong local anisotropy. On the contrary, close valued eigenvalues, $\lambda_1(n) \simeq \lambda_2(n)$, indicate that there is no dominant directional tendency of the gradient, meaning that the local texture patterns is roughly isotropic. The anisotropy indicator, $\eta(n)$ called also coherence in the literature [Weickert 2014] is given by:

$$\eta(n) = \frac{\lambda_1(n) - \lambda_2(n)}{\lambda_1(n) + \lambda_2(n)}. \tag{2.7}$$

The coherence can be seen as the confidence associated to the local orientation estimation. Its value varies between 0 and 1. A value close to 0 is an indicator of a low accuracy in the orientation estimation. The textural pattern corresponding to the local analysis window has no privileged orientation. In other words, the local textural pattern is either homogeneous, either composed of a mixture of different inter-winding local directional tendencies that define an isotropic character. On the other hand, a value close to 1 of the coherence parameter, associates a high confidence to the estimated local orientation.

It should be noted that a texture's local anisotropy, evaluated by the structure tensor's η parameter, does not imply the texture's global anisotropic character. More precisely, a texture can have an anisotropic character at a local analysis scale but at the scale of the whole texture have an isotropic character. Let us imagine an LST field composed uniquely of structure tensors with a strong difference between their eigenvalues. Although each structure tensor in the LST field can individually describe an anisotropic phenomenon, the whole LST field may be isotropic, if the structure tensors in the field have all different orientations. In this case, although the LST field is composed uniquely of anisotropic structure tensors, corresponding to locally anisotropic texture patterns, the global character of the texture is isotropic. Textures composed of circular patterns belong to this category, such as the cross section of a tree trunk. In this example, the texture is isotropic, although

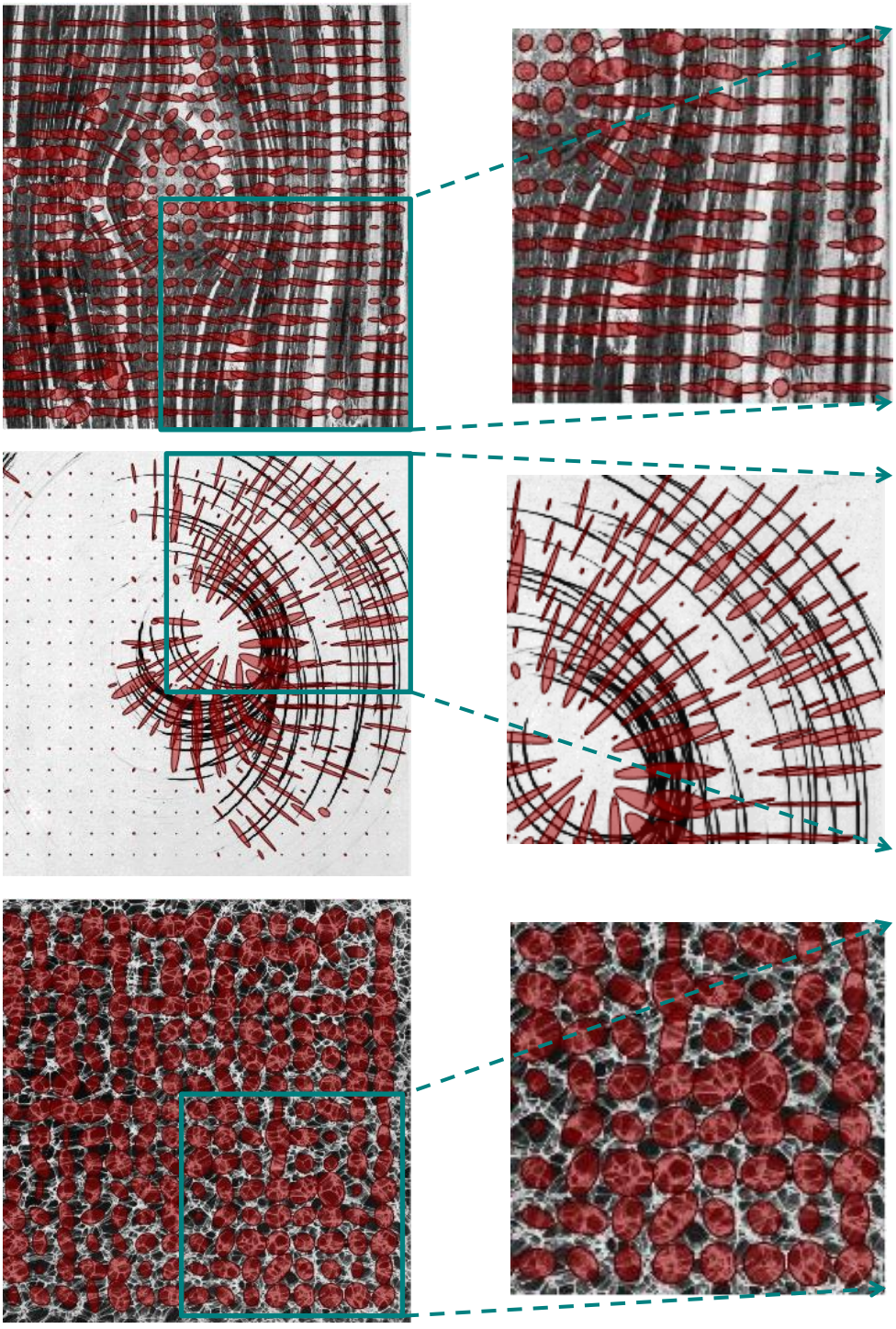


Figure 2.5: Ellipse form representation of the LST fields corresponding to the Brodatz database texture examples in Figure 2.4. The original size textures and zoomed areas are illustrated on the left and right columns, respectively. For each texture (from top to bottom), a sampling step equal to 28, 32 and 40 pixels, was used for displaying the ellipsoids.

locally, its corresponding structure tensors are anisotropic.

The structure tensor's eigenvalues can be exploited for assessing a texture's local energy, too. Computed as the sum of $\lambda_1(n)$ and $\lambda_2(n)$, it gives a measure of the gray level dynamics, or in other

words, of the local intensity contrast.

Exploiting the structure tensor information in every point of an image results in a field of parameters locally characterising a texture's content. The LST information can be represented by displaying the different parameters fields, as done in Figure 2.4 that illustrates the orientation, coherence and energy parameter fields for 3 textures of different degrees of anisotropy. The first texture is globally anisotropic. In addition to being characterised by high values of the η parameter, a small variation of the θ parameter value can be observed on the orientation map, except on the wood knot at the center of the image. The second texture is composed by regions alternating in anisotropy, i. e. a strongly anisotropic region of circular varying orientation and strong contrast (high values for the energy parameter) and an isotropic almost uniform region associated to low values for the coherence and energy parameters. The last example illustrates a globally isotropic texture. It is characterised by fast variations in the local orientation values, which gives the granular aspect of the orientation map. The map of the coherence parameter is characterised as well by a granular aspect given by the mix between isotropic and almost uniform areas inside the plastic bubbles and the anisotropic areas of circular varying orientation corresponding to the bubbles' edges. The fast local variation from low contrast regions particular to the regions inside the bubbles and hight contrast area specific to the edges of the bubbles is observed on the energy parameter map.

An alternative way of visually representing structure tensors is by means of an ellipse, as previously showed in Section 1.5 of Chapter 1. Examples of ellipse form representations of the LST fields are given in Figure 2.5, for the same Brodatz examples previously used for visualising the LST field's parameter maps. The ellipse form representation and the parameter maps representation of the LST field correspond to the same values of the structure tensor parameter's (σ_G equal to 1 and σ_T equal to 4, 5 and 4 respectively, considering the textures in Figure 2.5 from top to bottom). As it can be observed, the mainly anisotropic pattern of the first texture is characterised by thin and elongated structure tensors. As for the wood knot in the middle of the image, its isotropic character is described by close to circular ellipse forms, i.e having similar axes lengths. In addition, the medium size ellipse forms of the wood knot area are a mark of its low to medium varying energy levels.

The mixture of strongly anisotropic and isotropic patterns of the second texture is well characterised by ellipse forms varying in shapes and sizes. On the anisotropic regions of the texture, the ellipse form are more thin and elongated than in the case of the first texture, indicating a stronger degree of local anisotropy. As for the isotropic region, the close to circular ellipse forms are of small size, mark of the almost uniform pattern, as opposed to the isotropic region of the previous texture, where the higher level of local variation in the intensity values has been characterised by larger-size ellipses.

As far as the third texture is concerned, the ellipse form LST field is composed of medium-sized ellipses of low degrees of asymmetry and small ellipse form, characterising the uniform pattern inside the plastic bubbles. However, the fast variations in the texture's local characteristics are difficult to be visualised by means of ellipse form LST representation. A more dense representation

of the structure ellipse form would be necessary which would prevent a clear visualisation since neighbouring ellipses would be superposed.

In conclusion, the LST field information can be exploited by means of eigenvalues decomposition of the structure tensor computed in every point of the texture. The orientation, anisotropy and the local energy degree can be visually represented in different ways. The LST parameters maps representation can provide a dense illustration of the different parameters, in each point of the texture. However, one map per LST parameter needs to be generated. In contrast, the ellipse form representation has the advantage that each ellipse encompasses at he same time the orientation, anisotropy and energy information in the corresponding point. In addition, it can be superposed on the texture under analysis. On the contrary, due to the 2 dimensional shape of the object, in order to ensure a clear visualisation, i. e. that neighbouring ellipses are not superposing, a sparse representation needs to be applied. While for textures presenting slow variations in the local patterns geometry, the sparse representation (ellipse form LSTs) allows a relatively complete description of the LST field, in the case of textures with fast varying local characteristics, a dense representation (LST parameters maps) approach is preferable for a complete description of the texture.

2.2.3 LST for Multiscale Texture Analysis

A texture can be viewed as a scale-dependent data. Analyses conducted at distinct scales generally emphasize different characteristics. A low scale analysis is adapted for characterising fine details of the local patterns, while the information relative to the texture's global structure is lost. On the other hand, a large scale analysis leads to a more rough characterisation of the texture and gives the dominant elements of its geometric structure while small details (if present in the original image) are not characterised. Some of the textures are mono-scale, and their content can be well-characterised at a single scale of analysis. However, most real textures are characterised by a rich and diverse content and can be seen as multiscale data. This means that information relative to their characterisation is available at several different scales.

Given these considerations, a texture analysis method should be able to adjust its scale and perform the analysis at the texture scale containing the relevant information. This is the case of the structure tensor, that is a scale adjustable tool. For a same texture, the geometry information encompassed by the structure tensor at distinct scales will be different.

A first step for an accurate characterisation of the local orientation and anisotropy information is to identify the scale that is relevant for the texture information one wants to describe. Second, the size of the structure tensor's support should be adapted to match the scale of interest. This is easily done by tuning the 2 parameters of the LST, namely the parameters of the Gaussian derivative and weighting kernels, σ_G and σ_T respectively. The parameter σ_G should be chosen so that the gradient vectors capture the orientations of the local patterns. The choice of a small value for σ_G assures a local estimation of the gradient while increasing, at the same time, the noise sensitivity. The choice

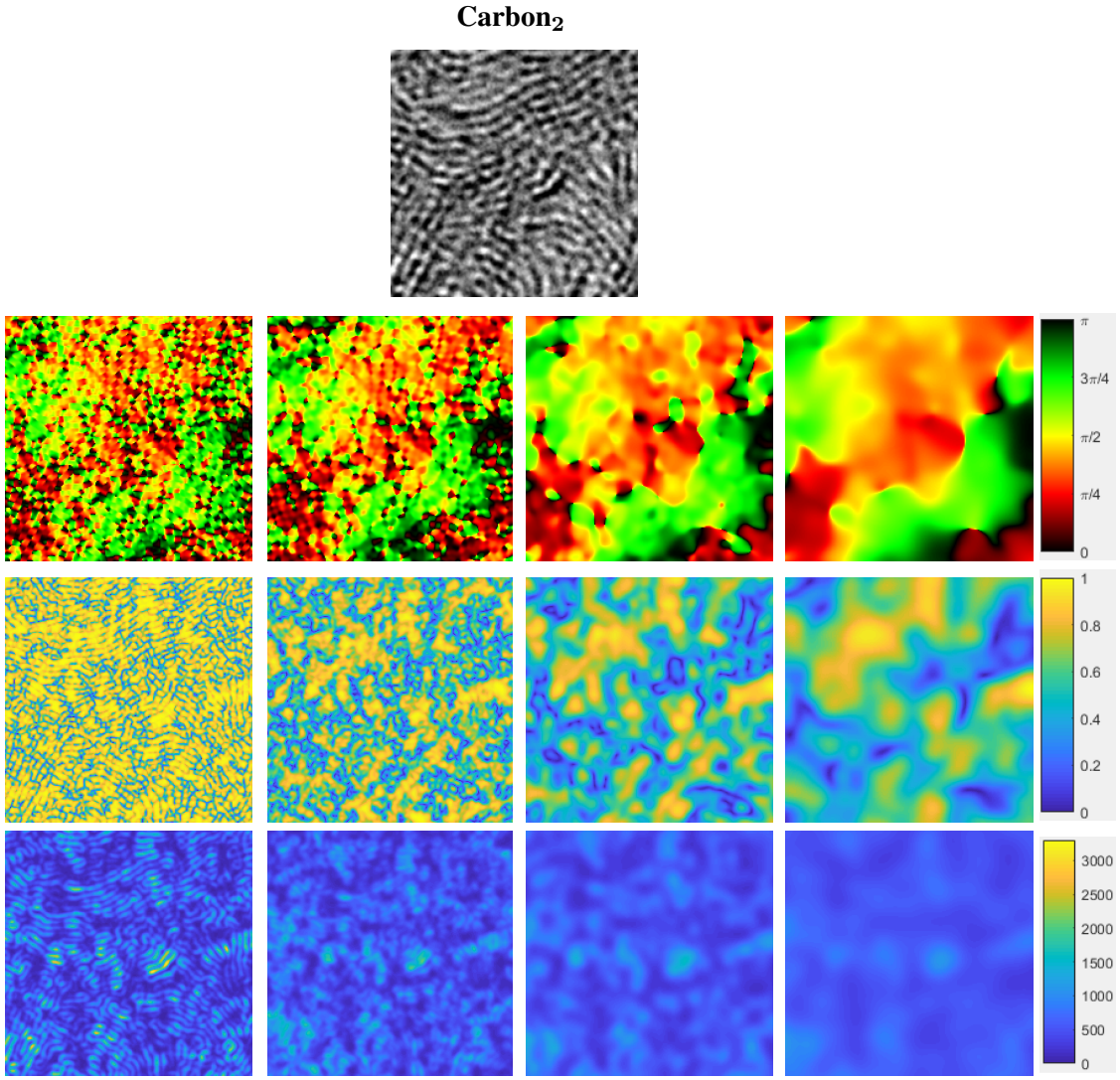


Figure 2.6: Analysis of an example texture of carbon material (first row) at four different scales going from high resolution to low resolution (from left to right). The LST field corresponding to each scale was computed for $\sigma_G = 1$ and a varying value of the parameter σ_T: $1, 2, 4$ and 8, respectively. Second to forth rows - LST orientation, coherence and energy fields, respectively.

of σ_T determines the size of the neighborhood within which the gradient vectors are pooled together to compute a directional tendency. A large value leads to a low-resolution analysis of the gradient field while a low value allows a finer scale analysis at the price of an increased noise sensitivity. As opposed to the σ_G parameter, a larger value is a better choice for σ_T. While it increases noise robustness, it ensures as well the homogeneity of information inside a window. For an accurate analysis, the size of the weighting window should be approximately the same as the size of the texture pattern of interest [Toujas 2010, Da Costa 2012, Akl 2015].

Figure 2.6 illustrates a texture at 4 different scales of analysis and the corresponding LST parameters fields (orientation, coherence and energy) for each scale. It can be seen that the visual textural information differs from scale to scale, increasing in details as the scale of analysis becomes finer. In order to adjust the analysis to the texture's scale of interest, the size of the Gaussian weig-

hting kernel was varied when computing the structure tensor. It should be noted that a low value for the structure tensor's parameter σ_T corresponds to a high resolution analysis while a high value for σ_T corresponds to a low resolution analysis of the textural pattern. As for the Gaussian derivative kernel, a constant value was chosen for the standard deviation, namely $\sigma_G = 1$.

For a value of the σ_T parameter equal to 1 (first row of Figure 2.6), we observe low energy and coherence values for the image areas corresponding to the ridges and valleys (regions of the image where the intensity level reaches a local extremum in a certain direction) and high values for the areas corresponding to the edges. This indicates that the analysis is too local and provides information at a scale that is lower than the size of the patterns of interest. As a consequence, values of σ_T larger than 1 should be employed. For larger values of σ_T, namely $2, 4$ and 8 (second to fourth columns of Figure 2.6), this phenomenon is no longer reproduced. For $\sigma_T = 2$ the LST yields a local analysis of the texture at a scale corresponding to the pattern size. Fast variations in the values of the LST parameters are observed on the LST parameters maps, in particular on the orientation map. An analysis of the fine details of the textural pattern, i.e. a high resolution analysis is achieved for a small values of σ_T. For a higher value of the weighting kernel parameter, namely $\sigma_T = 4$, quasi-uniform areas are observed on the LST parameter maps that correspond to the compact carbon layers of same orientation that are visible on the input texture. For a higher scale analysis, namely for $\sigma_T = 8$, corresponding to a lower resolution analysis, the LST field is smoothed. Neighbouring compact regions, characterised by small differences in their orientations, coherence and energy levels are merged together. Thus, larger areas characterised by a higher degree of uniformity are observed on the LST parameter fields. This aspect is in particular marked on the LST orientation map. Local details of the textural pattern's structure are lost and the textural pattern is characterised in terms of its global and most prominent structural traits.

As previously described, the structure tensor is a local texture descriptor relatively easy to compute. Defined in each point as a 2×2 non-negative definite matrix, it encompasses the local orientation information as well as an uncertainty measure associated to it. Relying only on two parameters, σ_G and σ_T, it can easily be tuned to the textural content one wants to describe.

2.3 LST Statistical Modelling

In the previous section, it has been shown that the structure tensor allows the extraction and visual representation of different structural characteristics related to the anisotropy of the local patterns composing a textured image. In order to provide a quantified description of such a structure tensor field, statistical methods are considered for characterising distributions of LSTs. To this purpose, the non-Euclidean statistical models previously presented in Chapter 1 come as natural choices. The statistical models consist of a Riemannian Gaussian distribution defined on the affine-invariant metric space and of a multivariate Gaussian model on the log-Euclidean metric space. For both cases, their corresponding mixture models have been considered as well. These statistical models

are employed, in a first stage, for characterising marginal distributions of structure tensors. The fit of the theoretical distribution to the LST field of different textures is analysed in Section 2.3.1, for single and mixture models as well.

The log-Euclidean model and its corresponding mixture model can easily be extended to characterise joint distributions of structure tensors as opposed to the affine-invariant model that does not facilitate such an approach. Given the expansion capacities of the LE model, it enables the characterisation of neighbouring groups of structure tensors, encompassing thus, a texture's spatial information. The LE model and its corresponding mixture model can be adapted as well for characterising LST fields computed at different scales of analysis. A mixed approach encompassing the characterisation of spatial and multiscale information is proposed as well. The different approaches for statistically modelling joint LST distributions are described in Section 2.3.2.

As a remark, the LE multivariate Gaussian distribution and the Riemannian Gaussian distribution are adapted for characterising symmetric strictly positive definite matrices. However, in certain degenerated cases, at least one of the strucure tensor's eigenvalues can be equal to 0. In these cases, the structure tensor is not strictly positive definite. In order to address this issue, a shrinkage estimation of the structure tensor can be considered [Ledoit 2004], ensuring that it is always positive definite and well-conditioned.

2.3.1 Statistical Modelling of Marginal LST Distributions

In a first step, the previously presented statistical models are applied for describing marginal LST distributions of textured images. In order to evaluate how well a theoretical model fits the LST field of a texture, the model's parameters are first estimated for the LST field. Second, a synthetic LST field is randomly generated following a statistical model whose parameters are equal to the parameter estimates of the real LST field. In this way, the distributions of parameters $\lambda_1, \lambda_2, \theta$ and η corresponding to the synthetic LST field can be represented and compared with the distributions of parameters of the observed LST field.

A set of textures characterised by various types of patterns and properties has been considered, on a spectrum going from simple to complex. These textures belong to remote sensing and material application fields. The size chosen for the empirical LST field varies, according to the size of the texture and to the values considered for the structure tensor parameters that dictate the border effect, as previously described in Section 2.2.1. The size of the simulated LST field is chosen as a multiple of the size of the empirical LST field and it is greater than 1 million.

Before assessing the fit of different statistical models to the empirical LST distribution, one needs to make sure that the LST field has been computed at a scale that allows a proper characterisation of the textural pattern. This is done by tuning the structure tensor parameters σ_G and σ_T, as briefly explained in Section 2.2.1.

LST parameter distributions computed at different scales are illustrated in Figure 2.7, for an

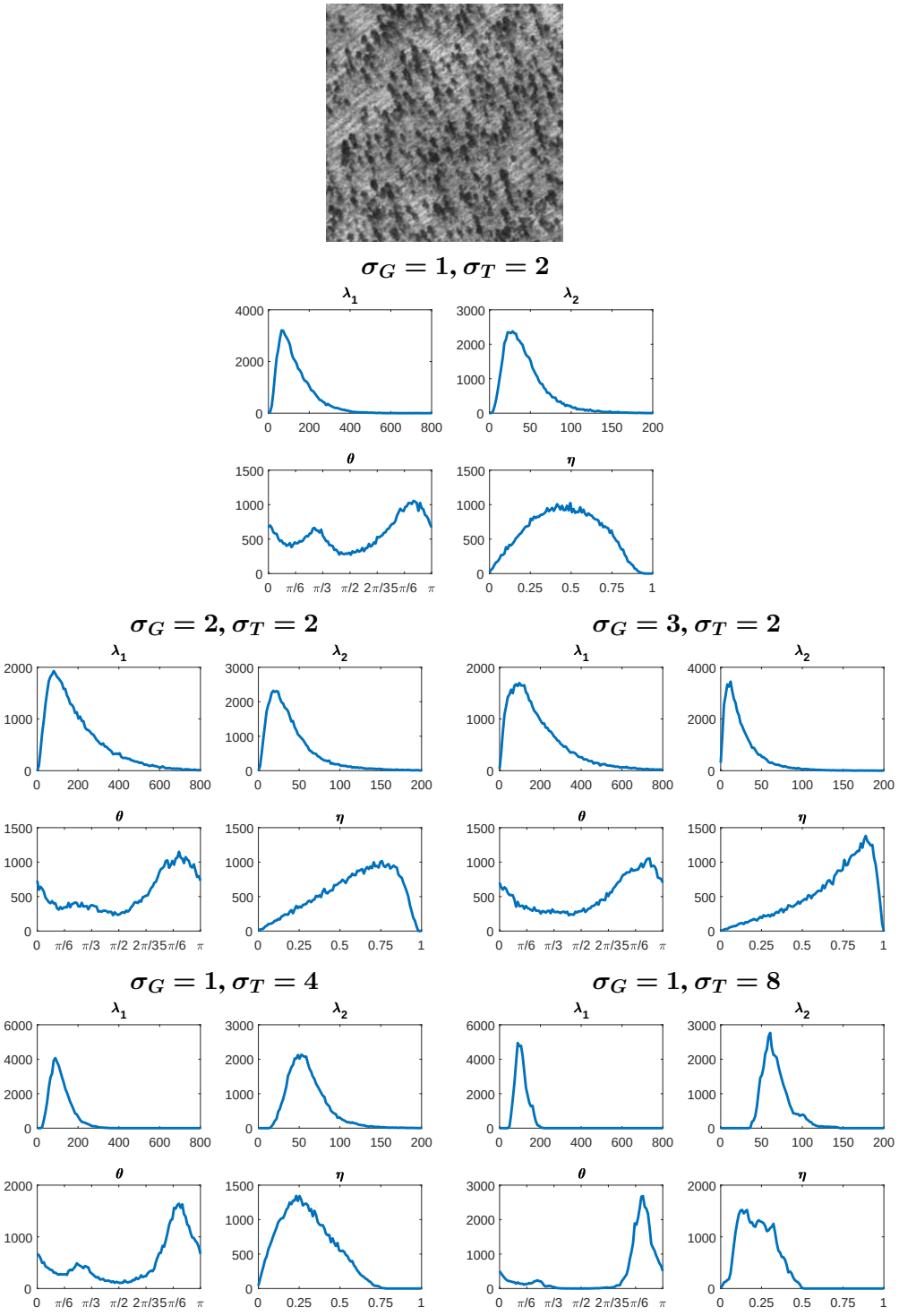


Figure 2.7: Distribution of LST parameters for different scales of analysis, for a remote sensing maritime pine forest texture. First row - correct scales of analysis for both the gradient and tensor parameters, namely σ_G and σ_T. Second row - variation of the gradient scale, σ_G. Third row - variation of the tensor scale, σ_T.

example remote sensing texture. The texture corresponds to a maritime pine forest stand of age greater than 20 years old. One can observe that visually, the texture is composed of inter-winding regions of locally anisotropic patterns - corresponding to the soil between the groups of trees - and

locally irregular and mostly isotropic patterns, representing groups of pines and the shadows they form on the ground. However, in spite of being locally isotropic, the trees displacement gives an average anisotropic character of the patten. Thus, the trees displacement in addition to the shadows they form on the ground give a first dominant gradient orientation that can be visually identified on the texture. The orientation's value ranges from $3\pi/4$ to π radians. A second dominant gradient orientation can be visually identified. It corresponds to the traces marked on the soil and its values is around $\pi/4$ radians. In addition to the dominant orientation, all the spectrum of orientations can be locally identified in the texture, as, for example, in the regions corresponding to isolated trees. In this case, the gradient corresponding to the areas around their crowns will vary in orientation, from 0 to π.

The LST parameters distributions should match the properties of the textural pattern identified at a visual inspection of the texture. For relatively small values of the parameters, namely $\sigma_G = 1$ and $\sigma_T = 2$, the LST parameters distributions reflect the visually identified properties of the texture. The two dominant gradient orientations are visible on the empirical orientation distribution that is bimodal. In addition, significant occurrences for all the values in the orientation spectrum are observed on the distribution. Furthermore, the average degree of anisotropy of the local pattern is well reflected by the empirical coherence distribution.

When fixing σ_T at its optimal value and varying the value of the derivative's filter standard deviation σ_G, it can be observed that the empirical distribution is degrading fast and it does no longer reflect the texture's characteristics identified at a visual check. The second dominant orientation of the gradient is no longer visible on the θ distribution that becomes uni-modal for $\sigma_G = 3$.

As for the standard deviation of the tensor's weighting filter σ_T, smaller degradations of the angle distribution are noticed as opposed to the previous case. However, for a value of $\sigma_T = 8$, the mode corresponding to the orientation of the soil traces is strongly attenuated. In addition, except from the gradient's dominant orientation, almost all the other local orientations are no longer visible on the empirical distribution. They have been eliminated as a result of a too strong smoothing. In addition, the coherence parameter takes lower values indicating a decrease in the confidence associated to the estimation of the local orientation. This is a natural consequence of an increased weighting window size that, in this case, includes neighbouring patterns that are different in terms of structure and orientation.

The results illustrated in Figure 2.7, show that the LST field is quite sensitive to the variation of the σ_G parameter. As a result, it is preferred to use a small value for the standard deviation of the derivative filter in order to account for the local variations of the textural pattern. In contrast, the LST field shows less sensitivity to the variation of the parameter σ_T. A larger spectrum of values can be tested in order to tune the weighing kernel to the size of the textural pattern of interest and to boost the LST's robustness to noise and other texture artefacts.

Once the LST computed at an accurate scale, the adjustment of the different statistical models to the empirical LST distributions can be analysed.

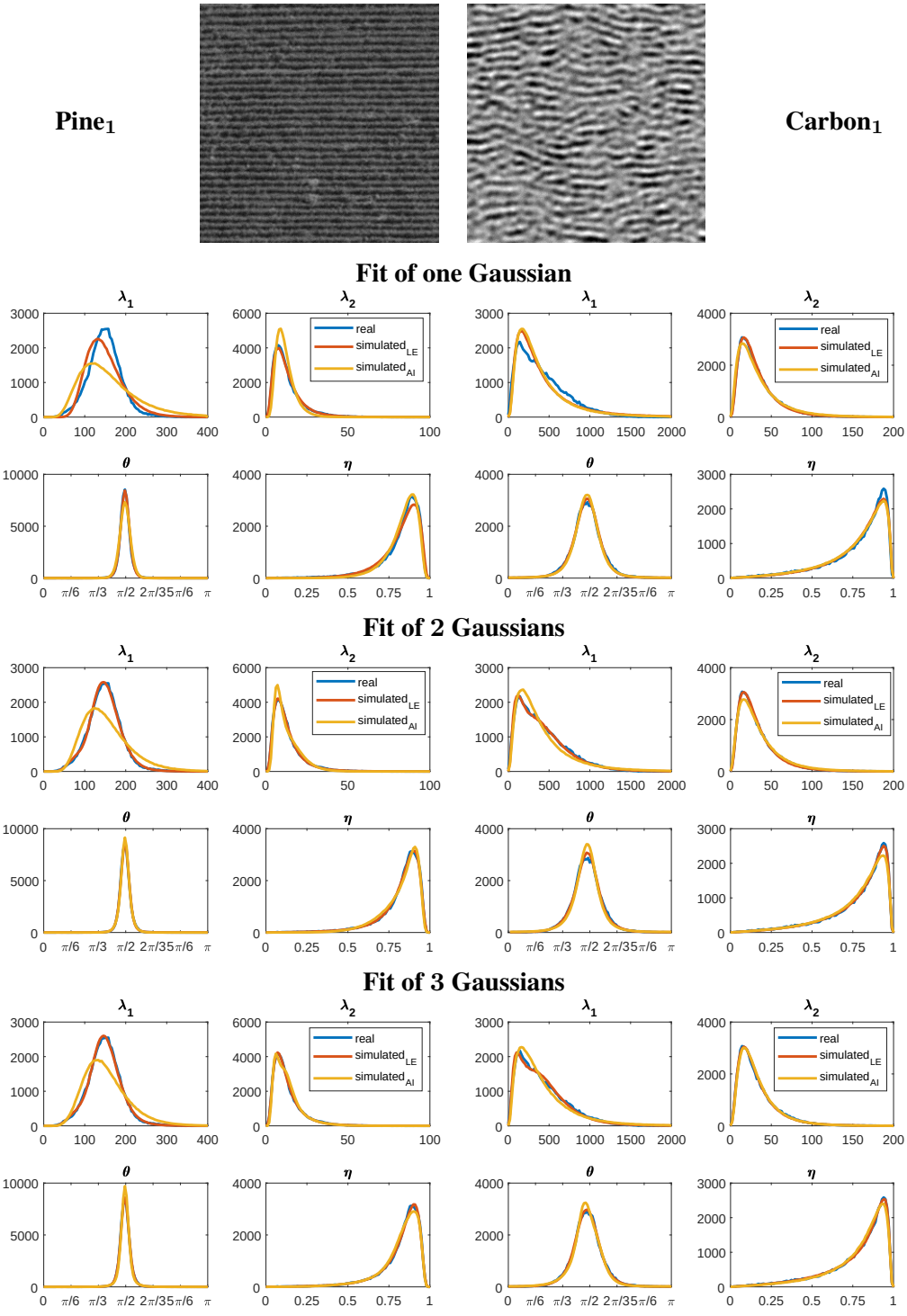


Figure 2.8: Adequacy of AI and LE statistical models for the characterisation of marginal LST distributions of higly structured anisotropic and mono-orientation textures. Comparison of the empirical distributions of LST parameters and the simulated distributions of LST parameters generated according to a multivariate Gaussian model on the LE space, a Riemannian Gaussian model on the AI space and their corresponding 2 and 3 components mixture models.

Figure 2.8 illustrates the distributions of parameters of the empirical and simulated LST fields of two textures characterised by relatively simple and regular patterns. The textures have a strong degree of anisotropy and one dominant orientation. The following values of the parameters have

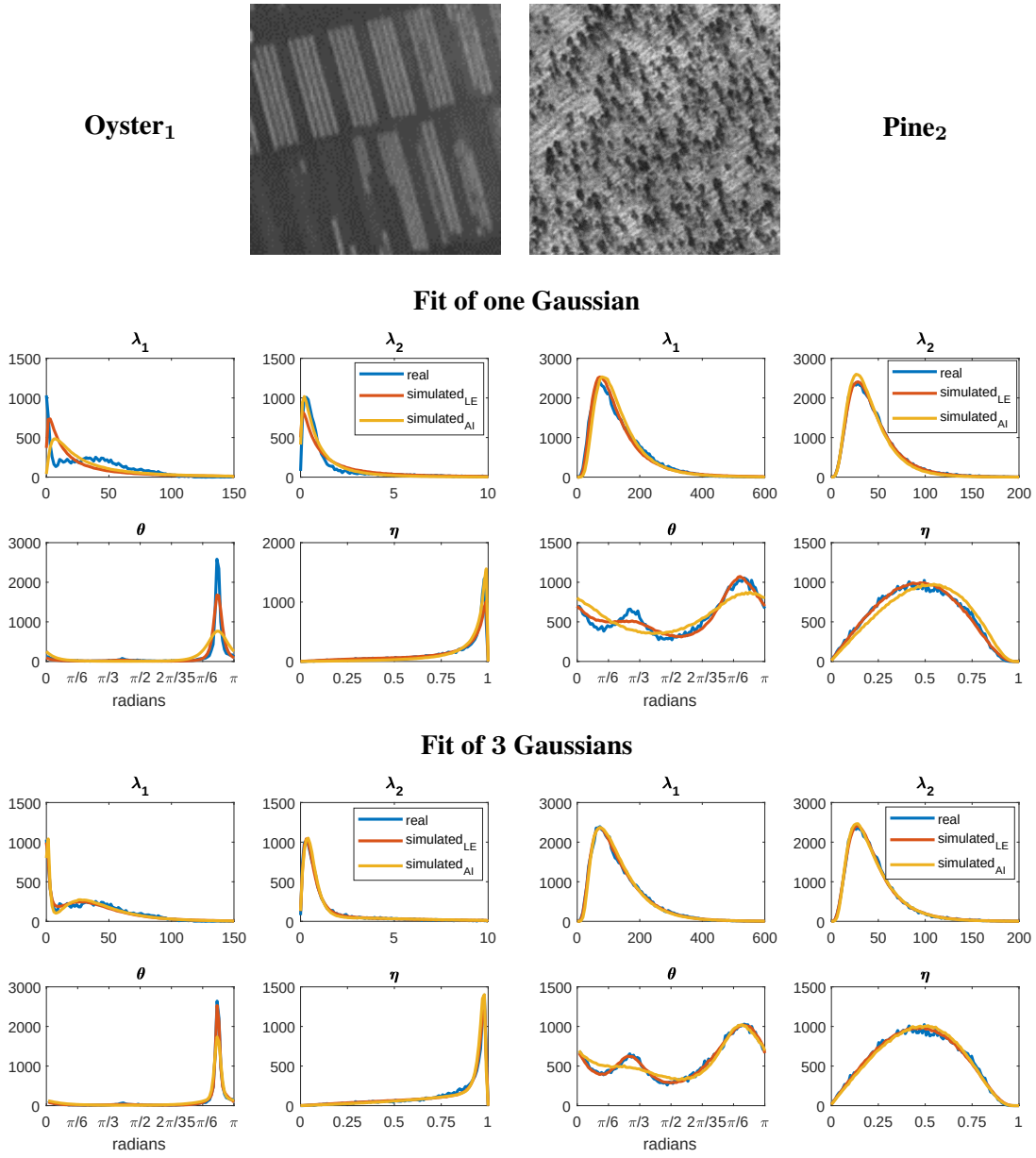


Figure 2.9: Adequacy of AI and LE statistical models for the characterisation of marginal LST distributions of anisotropic mono-orientation textures composed of alternating structured and unstructured regions. Comparison of the empirical distributions of LST parameters and the simulated distributions of LST parameters generated according to a multivariate Gaussian model on the LE space, a Riemannian Gaussian model on the AI space and their corresponding 3 components mixture models.

been considered when computing the LST field: $\sigma_G = 1$ and $\sigma_T = 2$.

Pine$_1$ illustrates a young remote sensing maritime pine forest of age less than 10 years old. It is a homogeneous and strongly anisotropic texture, aspects visible on the distributions of LST parameters: high values for λ_1 and low values for λ_2, the distribution of θ centered around $\pi/2$ and a high value of the coherence indicator (~ 0.9). As far as the statistical models are concerned, we can observe that for Pine$_1$, for a LE multivariate Gaussian model, the theoretical distributions of

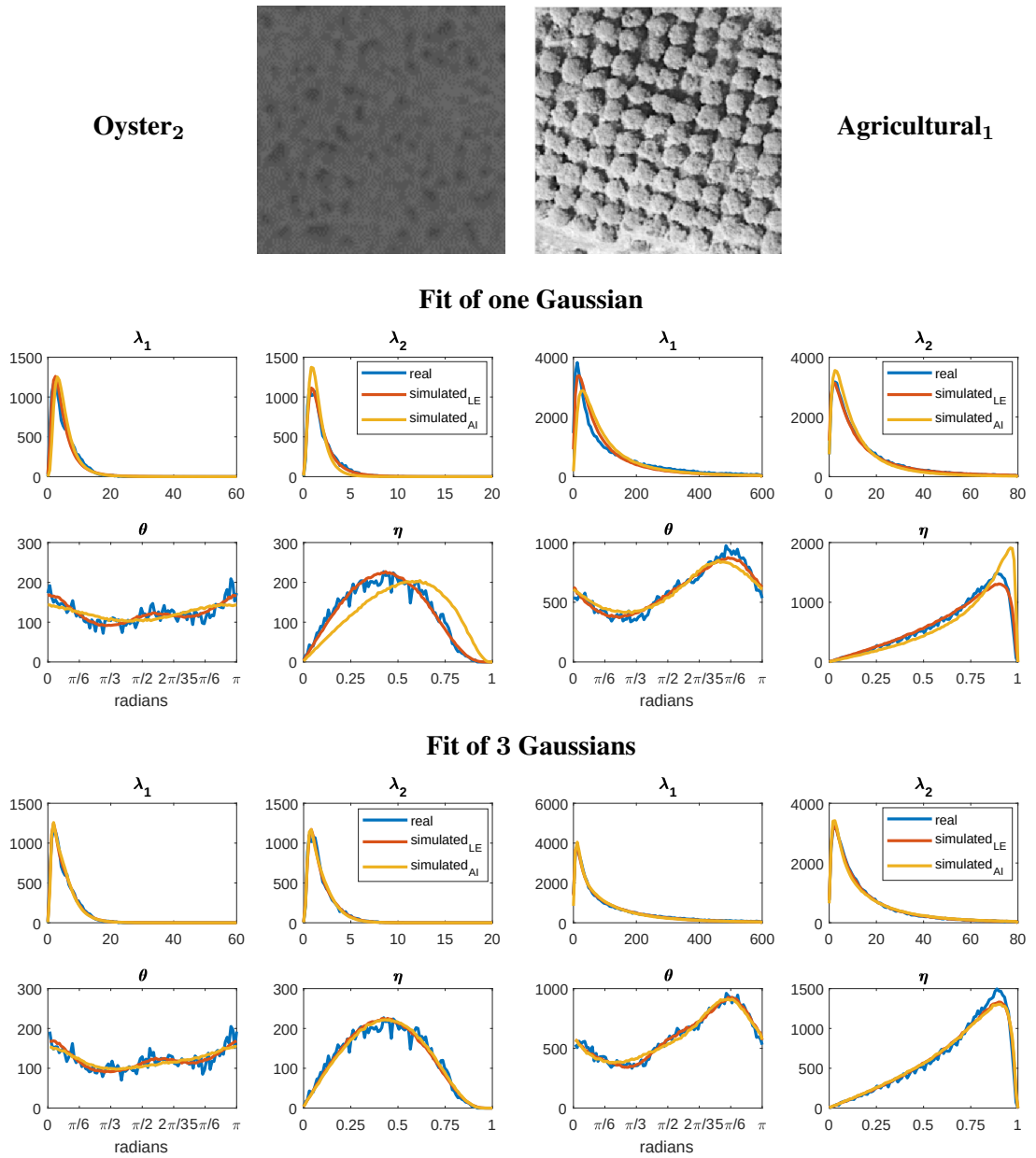


Figure 2.10: Adequacy of AI and LE statistical models for the characterisation of marginal LST distributions of isotropic textures. Comparison of the empirical distributions of LST parameters and the simulated distributions of LST parameters generated according to a multivariate Gaussian model on the LE space, a Riemannian Gaussian model on the AI space and their corresponding 3 components mixture models.

parameters fit well the empirical distributions of the LST field parameters, except for the distribution of λ_1. In addition, a better fit to the empirical data is observed for the LE model over the AI model. When it comes to the LE mixture model, it can be seen that all the theoretical distributions of LST parameters fit very well the empirical ones, starting for a number of mixture elements equal to 2. In the AI case, when employing a mixture model of 2 components, a slight improvement is observed in the adjustment of the theoretical distributions to the empirical ones, with respect to a single Gaussian model. However, the theoretical model is not fitting very well the observed data. When

an AI mixture model of 3 components is employed, a good fit is observed for most LST parameters. However, in the case of λ_1, it can be noticed that the theoretical distribution does not fit very well the empirical one.

Carbon_1 is a globally anisotropic texture, with the mode of the distribution of the coherence indicator close to 1. It is characterised by a dominant gradient orientation close to $\pi/2$ and small local variations around this value, as it can be observed on the distribution of θ. For this texture, all empirical distributions of parameters are unimodal and relatively simple. In consequence, both the LE multivariate Gaussian model and the Riemannian Gaussian one are fitting rather well the empirical data. Better adjustments are observed when mixture models of 2 and 3 components are employed, with a finer fit in the case of the LE mixture model. The increase in terms of characterisation precision of the statistical models from 2 mixture components to 3 is marginal.

Figure 2.9 shows the distributions of real and simulated LST field parameters of two textures composed of alternating anisotropic regions characterised by one dominant orientation and unstructured isotropic regions. Their LST fields are computed for the following values of the parameters: $\sigma_G = 1$ and $\sigma_T = 2$.

Oyster_1 is an oyster parks remote sensing texture illustrating several groups of cultivated oyster racks. They form regular and anisotropic patterns, of same orientation, alternating with the inter-racks regions whose pattern is isotropic and almost uniform. The mixture of anisotropic and isotropic patterns can be noticed on the distribution of λ_1 that is bimodal, one mode of very small values corresponding to the inter-racks regions and the second mode of higher values corresponding to the regions defined by the oyster racks. The texture's dominant orientation is given by the orientation of the oyster racks. The empirical distribution of θ is centred around this value and has a low variance. The texture is characterised by very strong local anisotropy, as indicated by the distribution of the coherence parameter, whose mode is very close to 1. In the case of a single Gaussian model, neither LE nor AI models fit well the empirical data. The theoretical distribution of θ adjusts better the empirical one in the case of the LE model, while the opposite behaviour can be noticed in the case of the distribution of λ_2. However, when employing a mixture of 3 statistical models, both LE and AI models fit quite well the empirical data, with a slight better adjustment of the LE model, observed in particular for the distributions of the orientation θ.

Pine_2 corresponds to an older forest of age greater than 20 years old. This texture has a lower degree of homogeneity than Pine_1. In addition, it is composed of inter-winding regions of locally anisotropic patterns - corresponding to the soil between the groups of trees - and locally isotropic and irregular patterns representing groups of pines and the shadows they form on the ground. The distribution of local orientations is no longer unimodal. Two dominant local orientations are identified with relatively high occurrences associated to every value on the orientation spectrum. While all models provide a good fit for most of the LST parameters distributions, the bimodal empirical distribution of θ arises fitting difficulties for most models. However, for a mixture of 3 LE multivariate Gaussian distributions, the theoretical distribution of the orientation is well fitting the empirical

one.

Figure 2.10 illustrates the distributions of parameters of the empirical and simulated LST fields of two textures of globally isotropic patterns. The LST field was calculated for $\sigma_G = 1$ and $\sigma_T = 2$ for the texture Oyster_2 and $\sigma_T = 1$ for the texture Agricultural_1.

Oyster_2 is an oyster park texture illustrating the foreshore. It is characterised by a relative simple and regular isotropic pattern. The isotropic character is well illustrated by the distribution of the parameter θ that is almost uniform. The distributions of the LST parameters show that one LE multivariate Gaussian model fits very well the observed data. This is not the case for the Riemannian Gaussian model, where misfits can be observed on the distributions of λ_2, θ and η as well. However, when employing a mixture of 3 Riemannian Gaussian models, the theoretical distributions fit well the empirical ones, except for the θ distribution that still does not adjust very well. While for this texture a mixture of Riemannian Gaussian models is necessary for a complete characterisation of its content, one single model is enough on the LE space. In addition, it seems to adjust even better than the mixture model on the AI space. This aspect can be observed in particular on the distributions of θ.

Agricultural_1 is a texture from the database Land Use Land Cover of Merced University. Its content is not specified, but given the shape and the arrangement of the trees in addition to their crown dimension, it might correspond to a poplar stand at maturity. The textural pattern is isotropic but the periodic displacement of the local patterns gives a global anisotropic character to the texture. The locally isotropic character of the patterns is reflected by the distribution of the parameter θ, characterised by high occurrences for all possible values of the gradient's orientation. One mode is observed on this distribution at a value close to $5\pi/6$. This orientation correspond to the shadows in between the tree crowns and it is interesting to notice that it does not corresponds to the orientation of the tree ranks that subjectively is perceived as being dominant, at a first visual check of the texture. As for the statistical models' abilities to characterise the empirical data, a good fit to the data is observed for a LE multivariate Gaussian model. In case of a mixture model of 3 components, the LE model fits almost perfectly the empirical data, with the AI model showing a slightly less precise fit, noticeable in particular on the orientations distributions.

So far, relatively simple textures composed of isotropic or anisotropic patterns of one dominant orientation have been considered. In the following, we will analyse the adjustment of the proposed statistical models for textures characterised by more complex patterns.

In Figure 2.11, the extent to which the LE and AI statistical models are fitting the LST field of a remote sensing agricultural texture is analysed. The LST field was calculated for the following parameters values: $\sigma_G = 1$ and $\sigma_T = 1$. The texture Agricultural_2 corresponds to a young poplar stand. It is composed of anisotropic patterns corresponding to the tree ranks and to the shadows they form on the soil, giving the texture's main orientation of value slightly below π. The inter-ranks space corresponds to an almost uniform pattern, i.e. isotropic. This mixture of anisotropic and isotropic patterns is visible on the distribution of λ_1 that is bimodal. The higher complexity

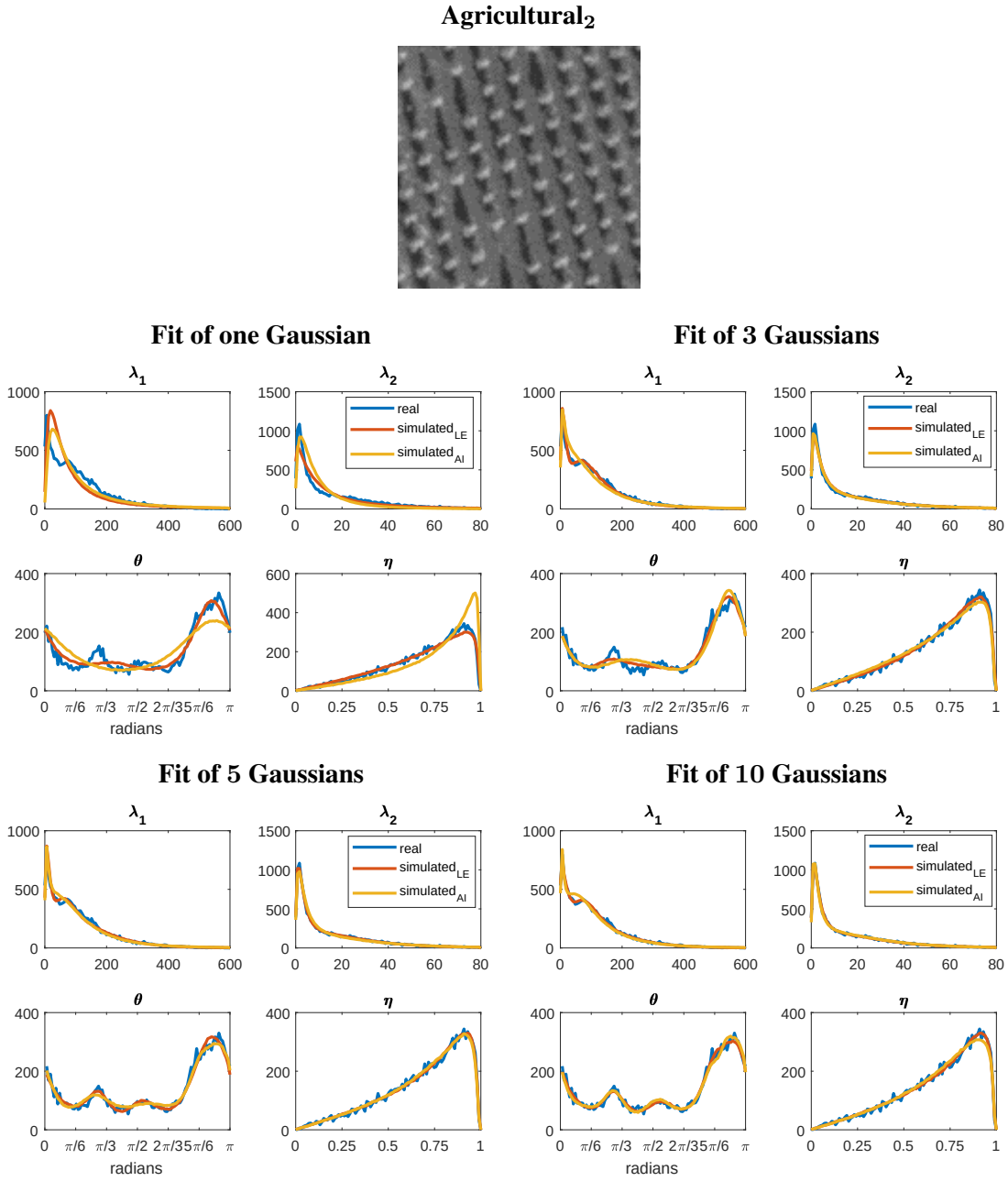


Figure 2.11: Adequacy of AI and LE statistical models for the characterisation of marginal LST distributions of a near-regular agricultural texture composed of alternating anisotropic and almost uniform patterns. Comparison of the empirical distributions of LST parameters and the simulated distributions of LST parameters generated according to a multivariate Gaussian model on the LE space, a Riemannian Gaussian model on the AI space and their corresponding 3 components mixture models.

of the textural pattern of Agricultural$_2$ with respect to the previously considered textures can be noticed as well on the orientation distribution that has 3 modes. While the multivariate Gaussian model is a better fit for the empirical data than the Riemannian Gaussian model, aspect observed in particular on the distributions of θ and η, the theoretical distribution of the λ_1 parameter does not adjust well to the empirical distribution. As a result, one Gaussian is not enough for a precise

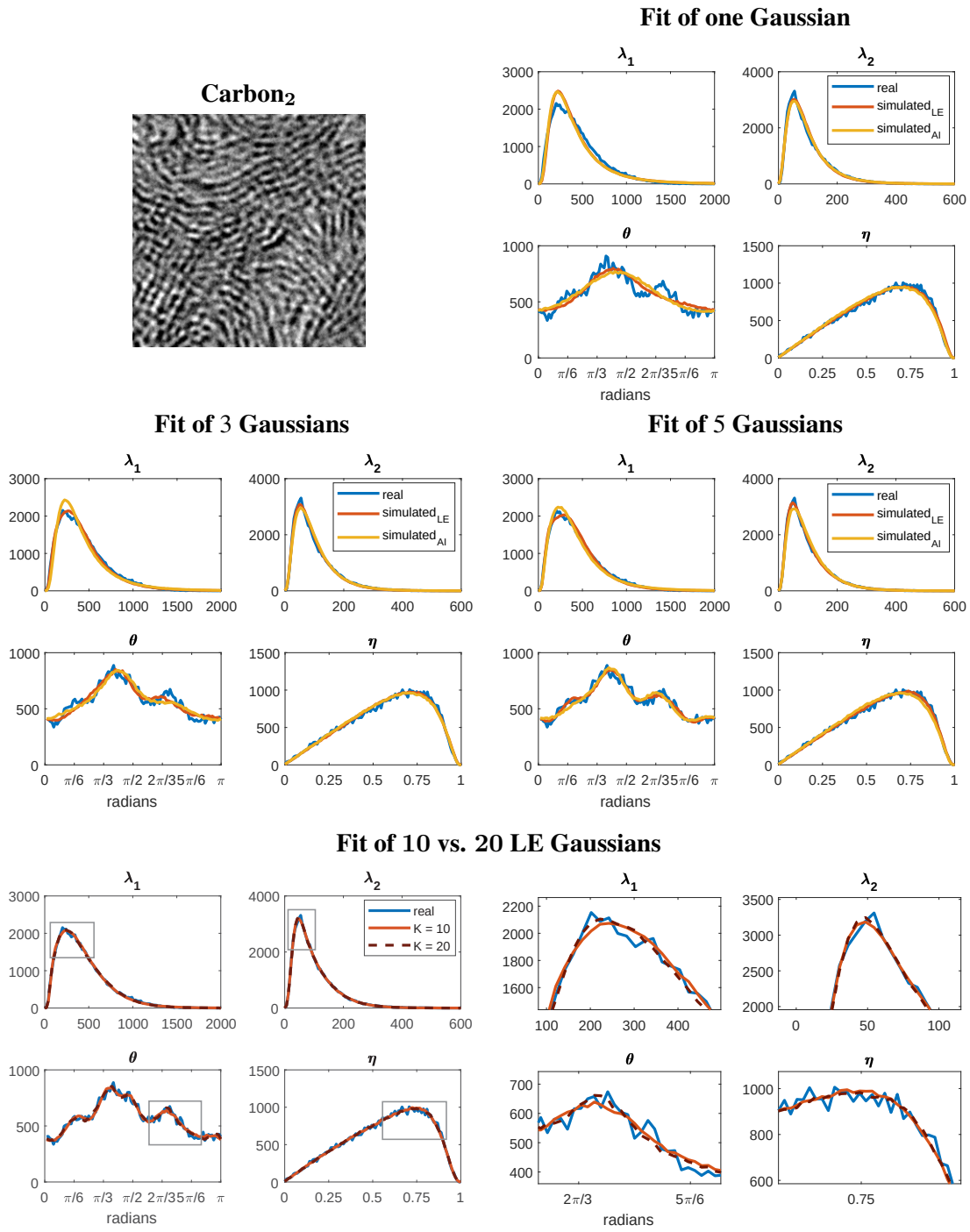


Figure 2.12: Adequacy of AI and LE statistical models for the characterisation of marginal LST distributions of a carbon material texture globally unstructured but locally anisotropic and structured. Comparison of the empirical distributions of LST parameters and the simulated distributions of LST parameters generated according to a multivariate Gaussian model on the LE space, a Riemannian Gaussian model on the AI space (first row) and their corresponding 3 and 5 components mixture models (second row). LE mixture models of higher number of components are illustrated on the third row (original and zoomed).

characterisation of the data. For a mixture model of 3 components, the LE theoretical distribution is fitting well the empirical distribution of λ_1. In contrast, in the case of the θ parameter, the theoretical distribution does not fit very well the empirical one. Since in both LE and AI cases mixture models

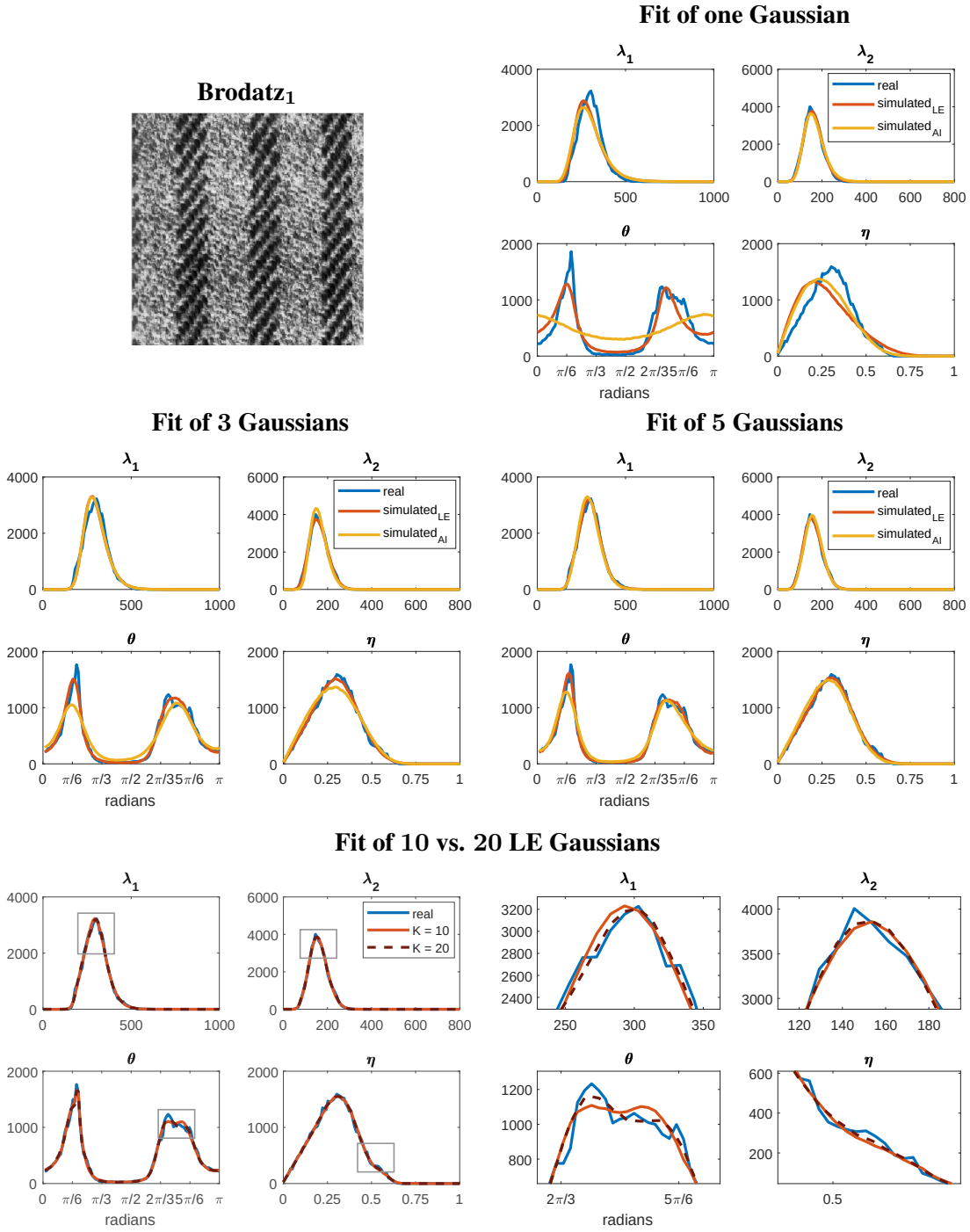


Figure 2.13: Adequacy of AI and LE statistical models for the characterisation of marginal LST distributions of highly structured texture of Brodatz database of combined isotropic and anisotropic patterns. Comparison of the empirical distributions of LST parameters and the simulated distributions of LST parameters generated according to a multivariate Gaussian model on the LE space, a Riemannian Gaussian model on the AI space (first row) and their corresponding 3 and 5 components mixture models (second row). LE mixture models of higher number of components are illustrated on the third row (original and zoomed).

of 3 components are not enough for a complete characterisation of the observed data, mixture models of 5 and 10 components are considered as well. While a very good fit is observed in the case of a LE mixture model of 10 components, its corresponding mixture model on the AI space adjusts less

well to the empirical data (aspect visible in particular on the distribution of λ_1).

In Figure 2.12, the fit of the proposed statistical models to the LST field of a carbon material texture is analysed. While being globally unstructured and characterised by a significant degree of randomness, Carbon_2 is characterised by a rich textural pattern that is locally composed of structured and anisotropic, slightly circular patterns. The diversity of local orientation is observed on the θ distribution, presenting high occurrences for all the values of the orientation spectrum and 3 modes. A relatively good fit is observed for the Gaussian models on both LE and AI metric spaces. However, the empirical distribution of the parameter θ is not well fit by the theoretical distributions. The mixture models of 3 components adjust slightly better to the empirical data, with the LE model being a better fit than the AI one. However, even in the LE case misfits can still be noticed, in particular on the orientation distribution. Slight adjustments of both LE and AI statistical models are observed for a mixture of 5 Gaussian distributions. Higher numbers of mixture components, i.e. 10 and 20 have been considered in the LE case. In both cases a better fit of the empirical distributions can be observed. In addition, while a mixture model of 20 components adjusts slightly better to the observed data than a mixture model of 10 components, the improvement is quite marginal.

Figure 2.13 illustrates the adaptability of the proposed statistical models for characterising the LST field of a Brodatz database texture. The texture is highly regular, composed of a mixture of anisotropic (dark vertical stripes) and isotropic patterns (light vertical stripes). Misfits of the theoretical distributions to the empirical ones are observed for all the theoretical models tested and illustrated in Figure 2.13, including the LE mixture model of 5 components. These misfits are mostly pronounced in the case of the θ distribution. Mixture models of 10 and 20 components have been considered in the LE case. While slight improvements are noticed, the empirical distributions are not perfectly matched by the theoretical ones. In addition, as in the previous case, the gain in terms of characterisation precision of the 20 components LE mixture model over the 10 components LE mixture model is minimal.

2.3.1.1 Study of the Optimal Number of Mixture Models

In the previous experiments we have seen that the complexity of the statistical model employed for texture characterisation is particular to the textural pattern. One LE Gaussian model showed a good fit to the observed LST field of textures characterised by simple patterns, as, for example, in the case of the texture Oyster_2 in Figure 2.10. However, for most textures, mixture models of LE or AI Gaussian distributions are necessary for a precise characterisation. Naturally, as a textural pattern is more complex and diverse, a higher mixture order is necessary for a good fit of the statistical models to the observed data. In the previous experiments the number of components of a mixture model was hand-tuned. However, different methods can be employed in order to automatically find the optimal value of this parameter for the texture under analysis.

One way of finding the optimal value of the parameter K is by studying the likelihood variation

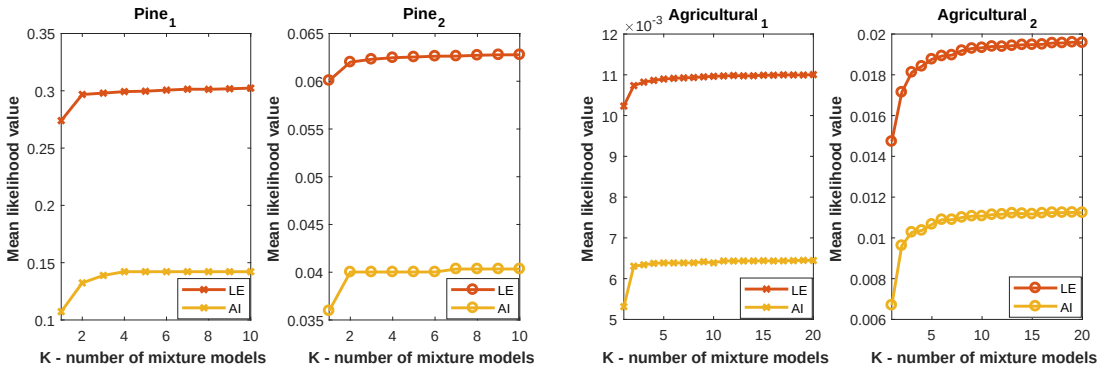


Figure 2.14: Variation of the mean likelihood value over all a texture's pixels for the LE and AI mixture models with respect to the number of mixture components, K. The variation curves correspond to the maritime pine forest textures illustrated in Figures 2.8 and 2.9 and for the agricultural textures illustrated in Figures 2.10 and 2.11.

for the considered statistical model according to the number of mixture components. In this section we analyse the variation of the mean likelihood value computed for all the structure tensors in a texture's LST field. The variation curves corresponding to the maritime pine forest textures Pine_1 and Pine_2 and for the agricultural textures Agricultural_1 and Agricultural_2 are displayed in Figure 2.14.

A relatively significant increase in the mean likelihood value of a mixture model over one Gaussian model is observed for the texture Pine_1, for both AI and LE statistical models. While for the LE models there is no noticeable increase in the mean likelihood value for values of K greater than 2, an increase is observed for the AI models for a value of K up to 4. The increase of mean likelihood value from a single Gaussian model to a mixture of Gaussian models on the LE and AI metric spaces is less significant for the texture Pine_2. However, it should be noted that a stable value of the mean likelihood does not imply that the statistical model is perfectly fitting the data. While for the LE mixture model of 3 components an excellent fit of the empirical distributions has been observed for the textures Pine_1 (Figure 2.8) and Pine_2 (Figure 2.9), the AI mixture model of 3 components is not fitting very well the empirical distributions. In other words, in particular for the AI statistical model, while increasing the number of mixture models often leads to a good fit of the empirical distributions of LST parameters, selecting the optimal value for the number of mixture components does not always guarantee a good fit of the empirical LST distributions.

When it comes to the agricultural texture Agricultural_1, the increase in the mean likelihood value for mixture models of 2 components with respect to a single Gaussian is marginal, for both LE and AI cases. In addition, for values of K greater than 2 the mean likelihood value is almost constant. These observations are in accordance with the illustratuins of LST parameters distributions in Figure 2.10 that showed a relativly good fit of both AI and LE theoretical models when one Gaussian was employed and a slightly better adjustment for a mixture model of 3 components. As for the texture

Agricultural_2 a stronger variation is observed in the mean likelihood value with respect to the number of mixture components, K. As opposed to the previous cases, a pronounced increase in the mean likelihood values is noticed for values of K up to 10. This confirms the results previously illustrated for this texture in Figure 2.11 that showed that mixture models of order higher than 5 are necessary for a good fit between the theoretical and empirical distributions of the LST field parameters.

Overall, the mean likelihood values for the LE models are greater than those corresponding to the AI models. However, the AI and LE mean likelihood values are not directly comparable as the data that is modelled belongs to different metric spaces. While the AI model is applied on the raw structure tensor data, the LE model is applied after transforming the original structure tensors data. We remind here that the transformation consists of computing the matrix logarithm of the tensor data and then vectorising the result. In consequence, if, for instance, one multivariate Gaussian distribution on the LE space has a higher mean likelihood value than a mixture model of Riemannian Gaussian distributions, it does not imply that the LE model is better fitting the data. However, the increase in the mean likelihood value of a mixture model over a single model is more pronounced for the AI case. In other words, a mixture model on the AI space is expected to provide better fit to the empirical data than a single Riemannian Gaussian distribution. In contrast, while employing a mixture model over one single multivariate Gaussian on the LE space improves the adjustment of the theoretical model to the empirical data, the gain in terms of characterisation precision is less marked than on the AI space.

In addition to the likelihood variation, *BIC* (Bayesian Information Criterion), introduced by Schwartz [Schwarz 1978], is commonly employed in model selection for automatically tuning the number of mixture components. The optimal value of the K parameter corresponds to the global minimiser of the *BIC* criterion:

$$\hat{K} = \arg\min_{K} BIC(K). \tag{2.8}$$

While selecting the number of mixture models according to the likelihood values remain a valuable method, this approach might lead to over-fitting. The *BIC* criterion is based on the likelihood function as well. However, it attempts to overcome the afore-mentioned limitation of the likelihood method, by adding a penalty term to the number of degrees of freedom of the model:

$$BIC(K) = -\mathcal{L} + \frac{1}{2} \times DF \times \log N, \tag{2.9}$$

where $\mathcal{L}$ and DF stand for the model's log-likelihood function and for the number of degrees of freedom, respectively. N corresponds to the sample set size.

In the case of a mixture model of Riemannian Gaussian distributions, the value of the log-likelihood function is computed according to equations (1.26) and (1.23). In the LE case, the log-likelihood function is computed by equations (1.51) and (1.49).

The number of degrees of freedom for the AI and LE mixture models are given in the following, for the case of sample sets of SPD matrices of size $m \times m$. The number of degrees of freedom for a

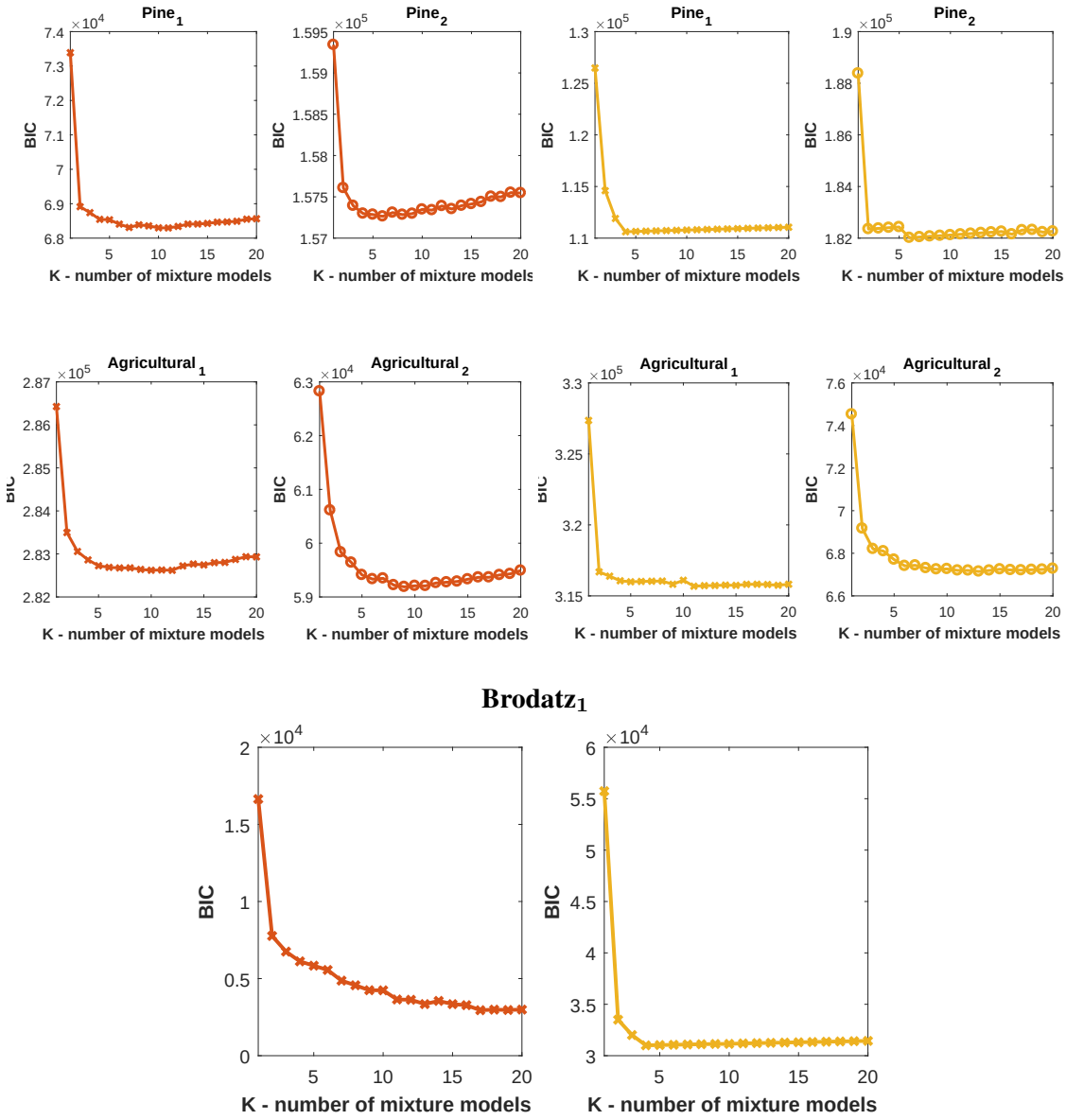


Figure 2.15: Variation the BIC criterion of the AI and LE mixture models with respect to the number of mixture component, K, for the maritime pine forest textures illustrated in Figures 2.8 and 2.9, for the agricultural textures in Figures 2.10 and 2.11 and for the Brodatz database texture in Figure 2.13. The BIC criterion variation curves corresponding to the LE statistical model are drawn in red while those corresponding to the AI statistical model are drawn in yellow.

mixture model of Riemannian Gaussian distributions is given by:

$$DF_{MG_{AI}} = K \times \frac{m(m+1)}{2} + K + (K-1), \tag{2.10}$$

with the following repartition of the degrees of freedom between the mixture model's parameters: the first term is associated to the centers of mass $\mathbf{M}_k, k = 1, ...K$, the second term corresponds to the dispersion parameters $\sigma_k, k = 1, ...K$ and the last term is associated to the mixture model's weights

$\omega_k, k = 1, ...K, \sum_{k=1}^{K} \omega_k$. The number of degrees of freedom for a mixture model of multivariate Gaussian distributions on the LE space is given as:

$$DF_{MG_{LE}} = K \times \frac{m(m+1)}{2} + K \times \frac{m(m+1)}{4}\left[\frac{m(m+1)}{2} + 1\right] + (K-1), \tag{2.11}$$

where the first term of the sum corresponds to the degrees of freedom of the mixture's models means $\boldsymbol{\mu}_k, k = 1, ...K$, the second to the degrees of freedom of the mixture model's covariance matrices $\boldsymbol{\Sigma}_k, k = 1, ...K$ and the last term is associated to the mixture weights of the statistical model, $\omega_k, k = 1, ...K, \sum_{k=1}^{K} = 1$.

The variation curves of the BIC criterion, according to the number of mixture components K, for both LE and AI models is illustrated in Figure 2.15, for several textures selected from the previous examples.

For textures characterised by relatively simple patterns, while a significant decrease in the value of the BIC criterion can be noticed from one Gaussian model to a mixture model of 2 components, marginal variations are observed for values of K greater than 2 as is the case for the textures Pine_1 and Pine_2. For these textures, the number of mixture components minimising the BIC criteria are low: 6 (LE case) and 4 (AI case) for the texture Pine_1 and 6 for both LE and AI models for the texture Pine_2.

Naturally, for textures characterised by more complex patterns, a stronger variation is noticed in the value of the BIC criterion for values of K greater than 2, in particular for the LE models. For instance, in the case of the texture Agricultural_2, the BIC criterion attains its minimum for value of K equal to 9 in the LE case and equal to 13 in the AI case. These observations are in accordance with the results illustrated in 2.11 where a good fit of both AI and LE theoretical models to the empirical data has been observed for a number of 10 mixture components. For textures presenting even more complex patterns, a more significant variation is observed for the BIC criterion corresponding to the LE case. This is the case of the texture Brodatz_1, for example, where the value of the BIC criterion is still in decrease for a value of K equal to 20. On the other hand, the value of K minimising the BIC criterion corresponding to the AI models is equal to 4. However, as previously illustrated in Figure 2.13, a mixture model of 5 components on the AI space does not provide a good fit of the empirical LST distribution.

Generally, it has been observed that in the case of the AI mixture models the BIC criterion attains its minimum for lower values of K than in the case of the LE models. However, in some cases, employing the optimal value for the number of mixture model components, as indicated by the BIC criterion, does not necessarily guarantee an excellent fit of the theoretical model to the empirical data.

The variation of the computation time, according to the variation of the number of mixture components, K, is displayed in logarithm scale in Figure 2.16. It corresponds to both LE and AI models applied on the texture Pine_1. Greater computation times are observed for the AI statistical

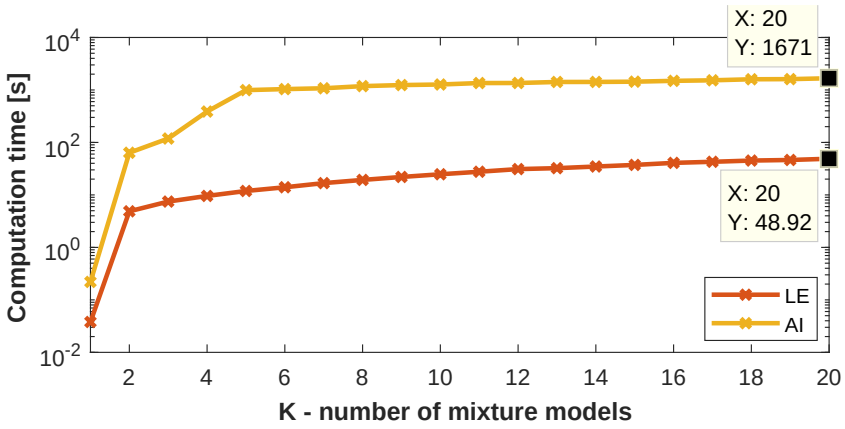


Figure 2.16: Logarithm scale variation of the computation time corresponding to the LE and AI mixture models with respect to the number of mixture components, K, for the maritime pine forest texture Pine_1 illustrated in Figure 2.8

models than for their LE counterparts. In addition, the computation time gap between the AI and LE models becomes more significant with the increase of the number of mixture models. Thus, if in the case of a single Gaussian distribution, the AI vs. LE computation time ratio is equal to 5.8, in the case of mixture models of 20 components, the LE model is 34 times faster than the AI one. More precisely, in this case, the model estimation takes around 50 seconds in the LE space and slightly more than 27 minutes in the AI case.

Generally, one LE multivariate Gaussian model fits quite well the empirical LST fields of relatively simple textures, for which the distributions of parameters are unimodal. A mixture of LE multivariate Gaussian models is necessary for fitting the LST fields of textures characterised by a mixture of patterns having different anisotropy and orientation properties. In these cases, the distributions of the LST parameters is, naturally, no longer unimodal. The previous experiments showed that a mixture of 3 LE multivariate Gaussian models is enough for a precise characterisation of most of the textures taken as examples in this section. However, in most cases, the improvement in the adjustment to the empirical data of a mixture model over one Gaussian model is not significant, on the LE metric space. For textures characterised by more complex patterns, a higher number of mixture components in necessary for a good fit of the theoretical model to the empirical data. When it comes to the AI model, the experiments showed that mixture models are necessary for well fitting the real LST fields of most textures. However, for some textures, even when increasing the order of the model until a maximum in the likelihood function is attained, the theoretical model is still mis-fitting the data.

A comparison of the LE and AI statistical models capacities to fit the LST fields of different textures can be summarised as follows. LE models globally adjust better to the empirical data than AI model of same orders. In some cases, a single LE multivariate Gaussian distributions showed a

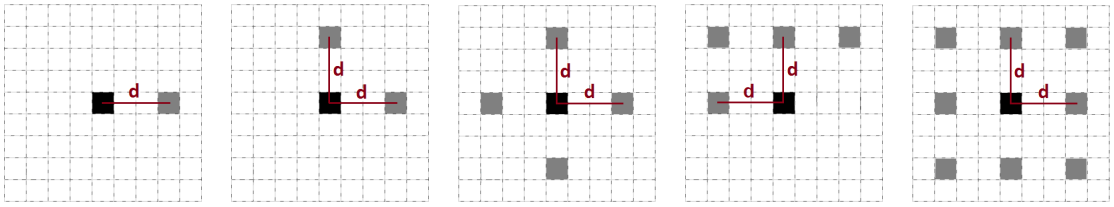


Figure 2.17: Examples of pixel neighbourhood configurations for an inter-pixel distance (d) equal to 3 pixels. In black - the current pixel, in gray - its neighbouring pixels. Configuration types, from left to right: couple, half-cross, cross - size 3, half-square - size 3 and square - size 3.

better fit to the empirical data than a mixture of 3 Riemannian Gaussian distributions, as in the case of the textures Oyster_1 in Figure 2.9 and Pine_1 in Figure 2.8. These differences are, on the one hand, natural, for models of same order, given the difference in the number of degrees of freedom between the LE and AI models. However, the experiments showed little gain in terms of the log likelihood function for AI mixture models of order higher than 5. In addition to showing a better characterisation potential of the LST fields of different textures, the LE statistical models are considerably faster than the AI models.

2.3.2 Statistical Modelling of Joint LST Distributions

While marginal LST distributions encompass the variability of the textural information in each point, the information relative to the spatial organisation of the textural pattern is lost. In addition, the previously presented approaches are performing the texture characterisation at a single scale of analysis. However, a texture is often composed of patterns defined at different scales. Thus, intuitively, a multiscale approach should be more adapted for texture characterisation. The Riemannian Gaussian model is not adapted for such an approach that implies modelling data samples of p-tuples of SPD matrices, where each p-tuple corresponds to a group of p neighbouring LSTs or LSTs computed at different scales. In contrast, the LE mapping, that allows the vector form representation of covariance matrices, makes possible to envision methods for statistically modelling joint LST distributions.

Building extended structure tensors encompassing the spatial, multiscale or both spatial and multiscale information in a given point of the texture can be easily achieved in practice, due to the LE vector form representation of structure tensors. However, as the dimension of the observed data increases, statistically modelling samples of extended structure tensors becomes a non-trivial task.

2.3.2.1 Extended LE LST Methods and Data Formatting

Three approaches for characterising p-tuples of structure tensors are proposed. They address the characterisation of spatial dependencies information, multi-scale information and the mixed description of spatial dependencies and multi-scale information.

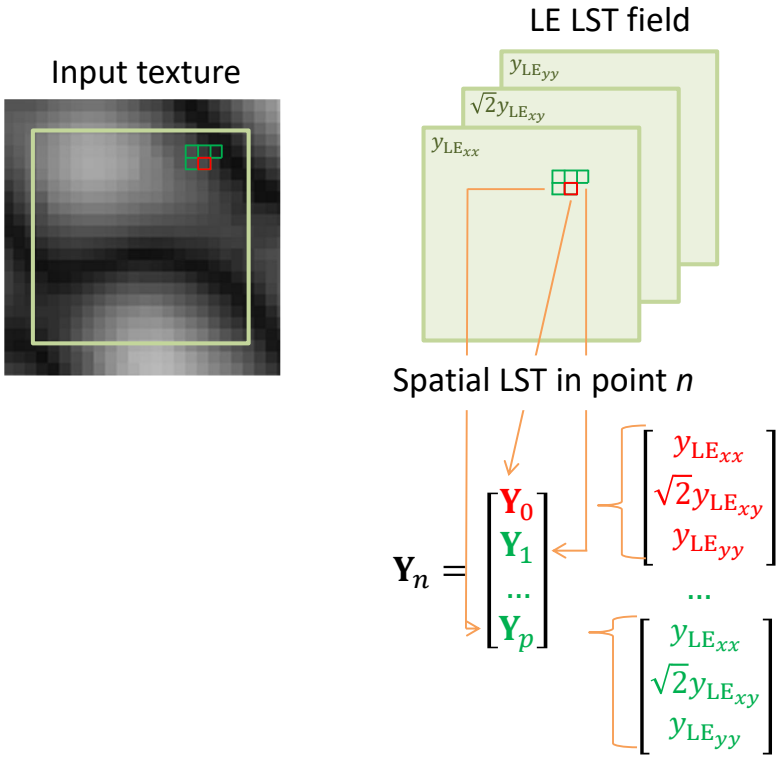


Figure 2.18: LE LST approach for describing spatial dependencies in an LST field. The border effect induced by successive filtering is drawn in light green on the input texture. The extended structure tensor at a point n, $n = 1, ...N$ where N corresponds to the LST field size, is formed by concatenating the LE vector form LSTs in the current pixel position and each point of the p size neighbourhood configuration. The countour of the current pixel is drawn in red and the neighbourhing pixels are marked in green. The same color code is used for the LE mapped structure tensors composing the extended structure tensor at point n.

Extended LST for describing Spatial Dependencies in LST Fields

In order to characterize the spatial dependencies inside an LST field, neighbouring structure tensors are grouped together, forming thus an extended structure tensor in each point of the texture. Several types of pixel neighbourhood configurations are considered with different values for the distance between the current pixel and its neighbours. Some examples of neighbourhood configuration for an inter-pixel distance equal to 3 are given in Figure 2.17.

Considering the LE vector form representation of structure tensors, given a neighbourhood configuration of p structure tensors, for each pixel, a vector is created by vertically concatenating the structure tensor corresponding to the current pixel and those in its neighbourhood, as illustrated in Figure 2.18. Thus, the size of the extended structure tensors in one point changes from 3 (as in the case of marginal LST distributions) to $3(p + 1)$.

The chosen neighbourhood configuration will induce a border effect. Thus, the extended LST distribution is defined uniquely for those pixels for which all neighbouring pixels' positions are defined within the texture's pixels positions domain.

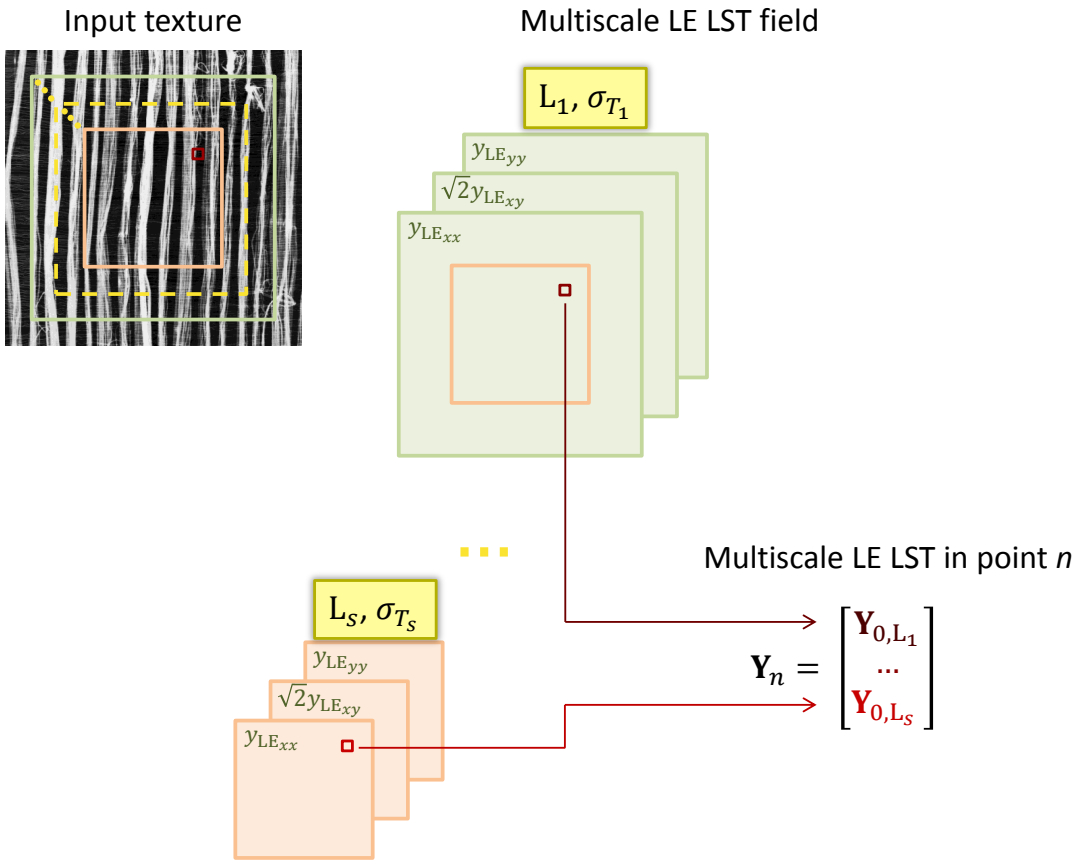


Figure 2.19: LE LST approach for describing multiscale dependencies in an LST field. The border effect corresponding to each scale $i, i = 1, ...s$, induced by successive filtering is marked on the input texture - in green for the scale L_1 and orange for the scale L_s. The extended structure tensor at a point n, $n = 1, ...N$ where N corresponds to the LST field size, is formed by concatenating the LE vector form LSTs at point n, for all scales $i, i = 1, ...s$. The contour of the current pixel is drawn in two nuances of red: dark (high resolution, scale L_1) and light (low resolution, scale L_s). The same color code is used for the LE mapped structure tensors composing the extended structure tensor at the point n.

Extended LST for describing Multiscale Dependencies in LST Fields

In response to a texture's character, a multiscale LST texture analysis approach is proposed, on the LE metric space.

First of all, the LST field is computed at different scales of analysis by fixing the value of the structure tensor's σ_G parameter and tuning the σ_T parameter. Each level of analysis, L_i, $i = 1, ...s$, where s denotes the numbers of scales considered, is associated a value σ_{T_i}. As the scale increases, the values of the σ_{T_i} parameter increases as well. However, the spatial resolution decreases with the scale increase. Thus, the scales succeed in decreasing order of spatial resolution from L_1 to L_s. Consequently, the size of the border effect associated to the LST field computation is specific to the scale and its value is maximal for the highest level L_s. So that a given pixel position has correspondence in all the levels of the multiscale analysis, the LST fields computed at scales $L_i, i = 1, ...s - 1$ are truncated to the LST field size of level s, as is illustrated in Figure 2.19.

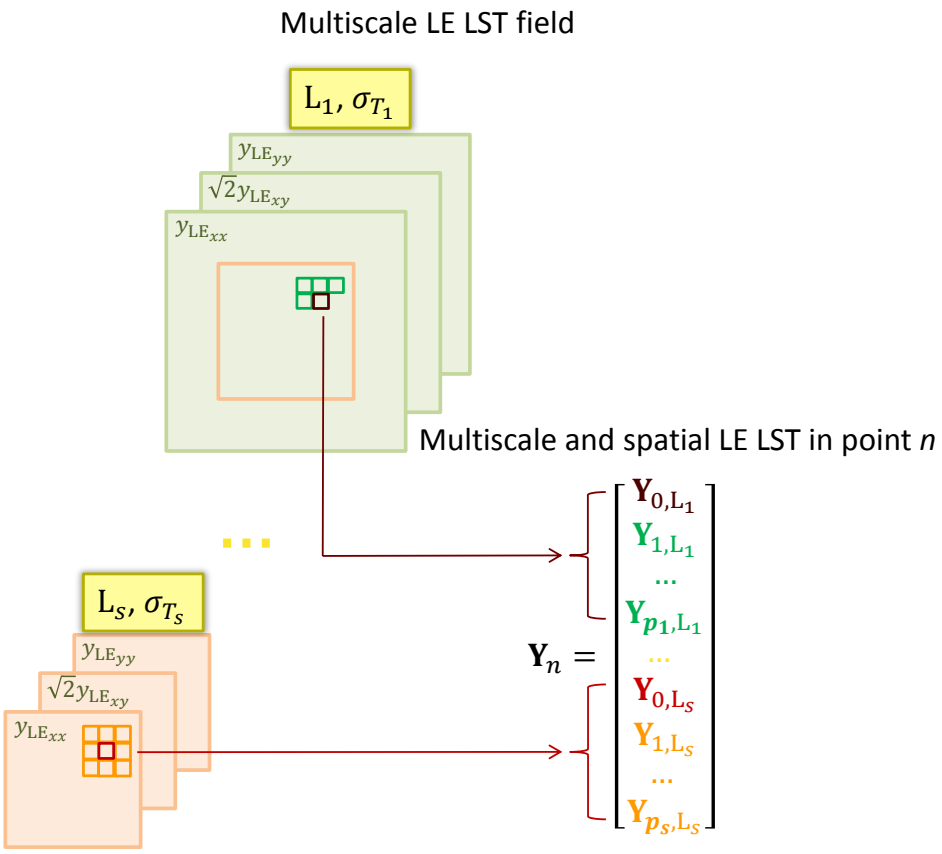


Figure 2.20: LE LST approach for describing spatial and multiscale dependencies in an LST field. The extended structure tensor at a point n, $n = 1,...N$ where N corresponds to the LST field size, is formed by concatenating the LE vector form LSTs at the current pixel position and each point of the p_i size neighbourghood configuration, for all scales $i, i = 1,...s$. The contour of the current pixel is drawn in two nuances of red: dark (high resolution, scale L_1) and light (low resolution, scale L_s). Different colors are used for the neighbourhood pixels at the two scales of analysis. The same color code is used for the LE mapped structure tensors composing the extended structure tensor at point n.

The multiscale approach consists in considering the structure tensors at each scale separately. Namely, the LE extended structure tensors at the point $n, n = 1,...N$, where N is the size of the LST field, is formed by concatenating the LE structure tensors at the point n, corresponding to all scales of analysis. In this case, the extended structure tensor is of size $3s$. A general scheme for this approach is illustrated in Figure 2.19.

Extended LST for describing both Spatial and Multiscale Dependencies in LST Fields

The previous approaches can be combined in order to develop an approach encompassing a texture's spatial and multiscale dependencies simultaneously.

At each scale, neighbourhoods of structure tensors are considered. Different neighbourhood sizes and configurations can be considered at each scale. The number of positions in the neighbourhood configuration at scale i is denoted by p_i. Thus, the extended structure tensor for this approach

is of size $\sum_{i=1}^{s} 3(p_i + 1)$. The general scheme for this multiscale LE LST based texture characterisation approach is illustrated in Figure 2.20. In addition, a mixed approach can be considered as well, where at some scales of analysis, the intra-scales LST dependencies are characterised by considering neighbourhoods of structure tensors and at some other scales, only the tensors on the current position are included when composing the extended structure tensors, without considering a neighbourhood configuration.

2.3.2.2 Statistical Modelling of Extended LST Distributions

Once the LE extended structure tensors are formed in each point, the resulting LST field is modelled by a LE multivariate Gaussian distribution or by a mixture of LE multivariate Gaussian distributions. The LE multivariate Gaussian distribution family is considered for all the afore-mentioned extended LE LST approaches (spatial, multiscale and the mixed spatial and multiscale approach). Although the model itself does not change, the dimension M of the LE multivariate Gaussian increases from 3 to $3(p + 1)$ in case of the mono-scale spatial extended tensors and to $\sum_{i=1}^{s} 3(p_i + 1)$ in the case of multiscale extended tensors, with $i = 1, ...s$ and s being the number of scales considered.

When the information at a different point (neighbour or different scale) is added to the information at the current point, there is an order 3 increase in the size of the extended structure tensor. Thus, there is a fast increase in the dimension of the observation space. An increase in dimension induces, on the one hand, a fast increase in the volume of the observation space. The sample size needed for covering the observation space in all the dimensions grows exponentially with its dimension [Kanevski 2009]. In consequence, high dimensional spaces *tend to be terribly empty* [Tarantola 2004], as the observed data becomes sparse. In addition, an increase in the dimension of the observation space leads to larger values for the number of degrees of freedom of the statistical model, i. e. a larger set of parameters that need to be estimated, as illustrated in Figure 2.21. Previously, we have observed that in 3 dimensional space the LE model has been more adapted than the AI one for characterising SPD matrices sample sets, in particular due to its higher number of degrees of freedom (see Section 1.5 of Chapter 1). While generally, a higher number of degrees of freedom of a statistical model amplifies its characterisation capacities, increasing the degrees of freedom above a certain limit becomes problematic, as a very large observation set is required for statistical inference.

The study of the theoretical models fit to randomly generated sample sets of extended structure tensors will be addressed differently here, as opposed to the previous case of marginal models. In the case of extended structure tensors, the covariance matrix structure of the structure tensors is lost, as they are formed by concatenating LE mapped vector form representation of structure tensors. Thus, the visual comparison of the theoretical and empirical parameters distributions (namely, λ_1, λ_2, η and θ), is no longer feasible. For marginal models, the size of typical LST fields corresponding to real textures allows to estimate the parameters of the 3 dimensional statistical model with a satisfac-

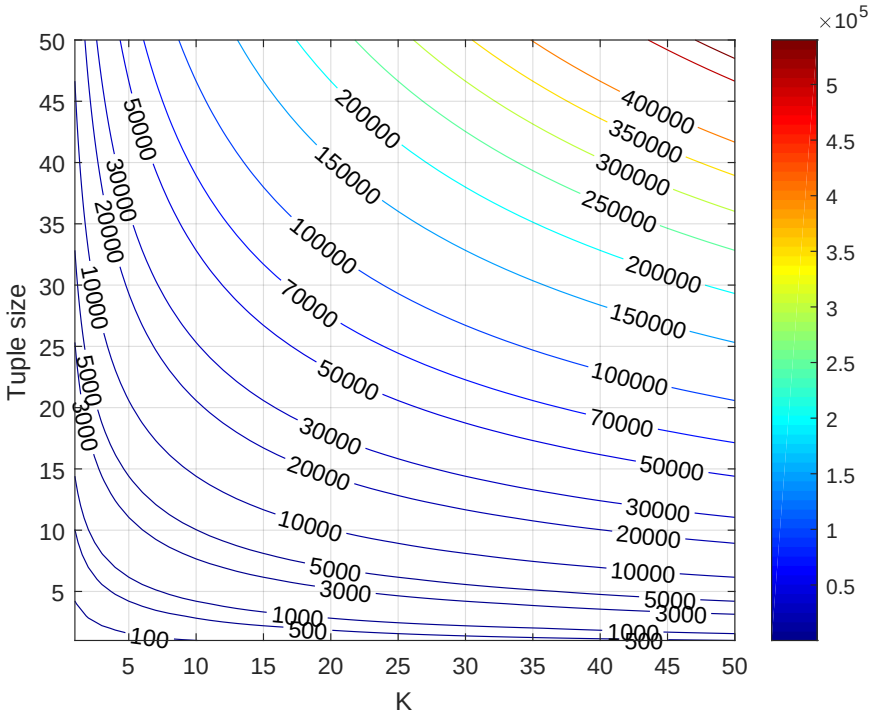


Figure 2.21: Variation of the number of degrees of freedom of a mixture of multivariate Gaussian distributions according to the structure tensor tuple's size and to the number of mixture components.

tory precision for the application purposes considered in this work. Most real textures used in our experiments are of size 256×256 which corresponds to an LST field of size generally higher than $50,000$, depending on the border effects. When modelling extended structure tensors distributions, as there is a dimensionality increase, the LST field size may not be sufficient for a robust estimation of the statistical model's parameters. In addition, considering mixture models further increases the already problematic dimension of the parameter space. The potential instability issues of the statistical inference process makes the study of the BIC criterion and of the likelihood variation according to the number of mixture models biased. The problem of statistical inference needs to be analysed in more detail. More precisely, the impact of the data sample size on the estimation of the statistical model's parameters in high-dimensional spaces needs to be assessed.

A test scenario is imagined in the following in order to analyse the impact of the sample set size on the likelihood function estimation, as the dimension of the observation space increases. To this purpose, sample sets of different sizes are drawn from a mixture of 3 multivariate Gaussian models. Four dimensions of the observation space are considered: 3 - corresponding to individual structure tensors, and $6, 9$ and 15 - corresponding to the following 3 structure tensors neighbourhood configurations: pair, half-cross and cross, respectively. The variation curves of the mean likelihood function, for sample sets sizes ranging from 500 to $1,000,000$ are illustrated in Figure 2.22. Naturally, the mean likelihood function diverges for small sample sets ($N = 500$ and $N = 1000$), no matter the dimension of the observation space. In a stable estimation case, the value of the likelihood

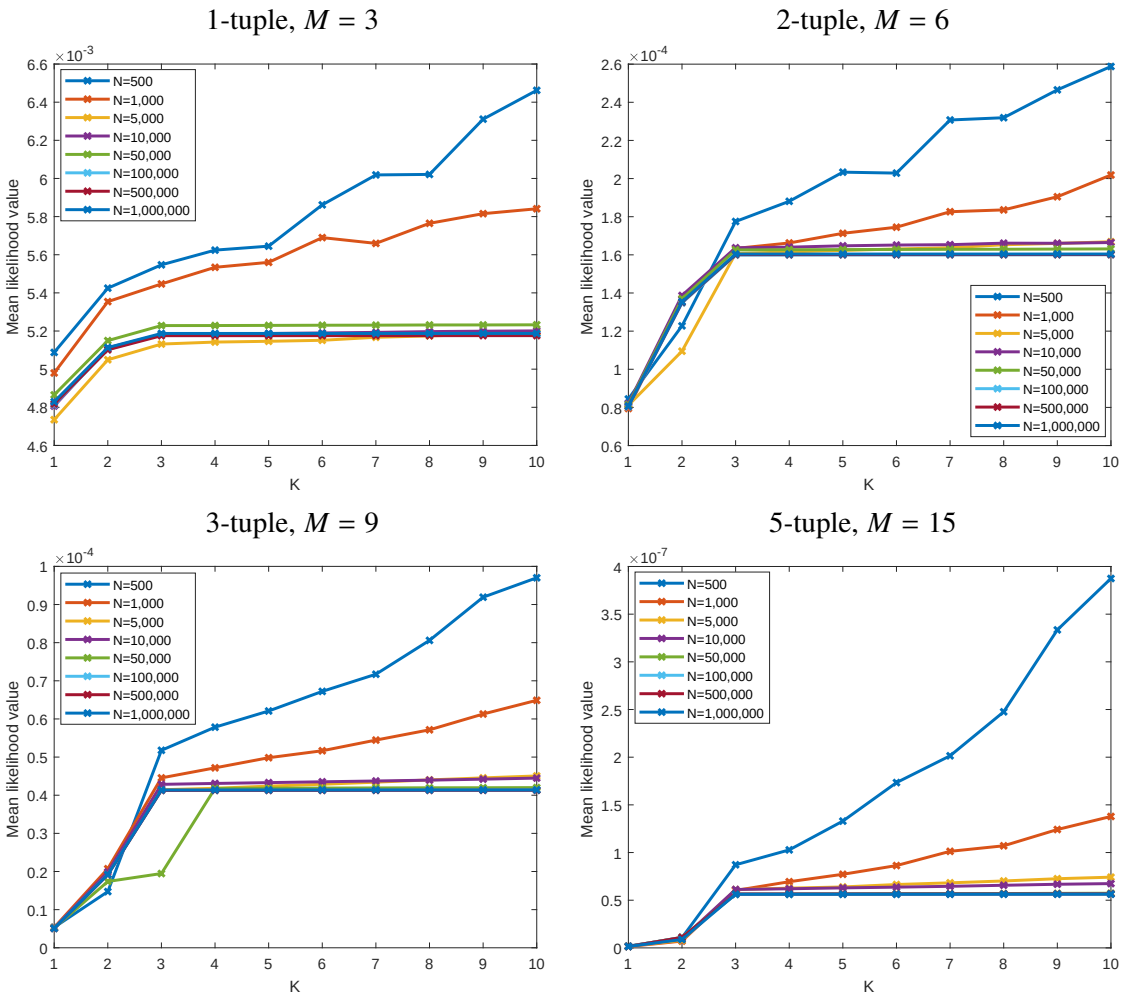


Figure 2.22: Variation of the mean likelihood value for simulated data samples of LE multivariate Gaussian models, according to the number of mixture components, K. The real number of mixture components of the simulated data is 3. The impact of the sample set size on the likelihood function estimation is illustrated. Four model dimensions are tested: $3, 6, 9$ and 15 corresponding to 1-tuple structure tensor, and to the following neighbourhood configurations: pair (2-tuple of structure tensors), half-cross (3-tuple of structure tensors) and cross (5-tuple of structure tensors), respectively. The neighbourhood inter-pixel distance is equal to 8.

function should increase until the real number of mixture components is attained and stabilize its value afterwards. Some degenerated cases are attained, when the dimension of the space is large and the number of mixture models is high. They correspond to cases when the number of degrees of freedom is larger that the sample set size. A relatively stable behaviour of the log likelihood function is observed for sample sets larger than $10,000$, in the case of observations of dimensions 3 and 6. When the dimension of the observation space is 9, unstable behaviour of the likelihood function is observed for larger sample sets, as well. For $N = 50,000$ the mean likelihood converges for a number of mixture components different than the real one (4 instead of 3). For an observation space of dimension 15, the divergence of the likelihood function for relatively small sample sets ($N = 5,000$ and $N = 10,000$) is more pronounced than in the previous cases.

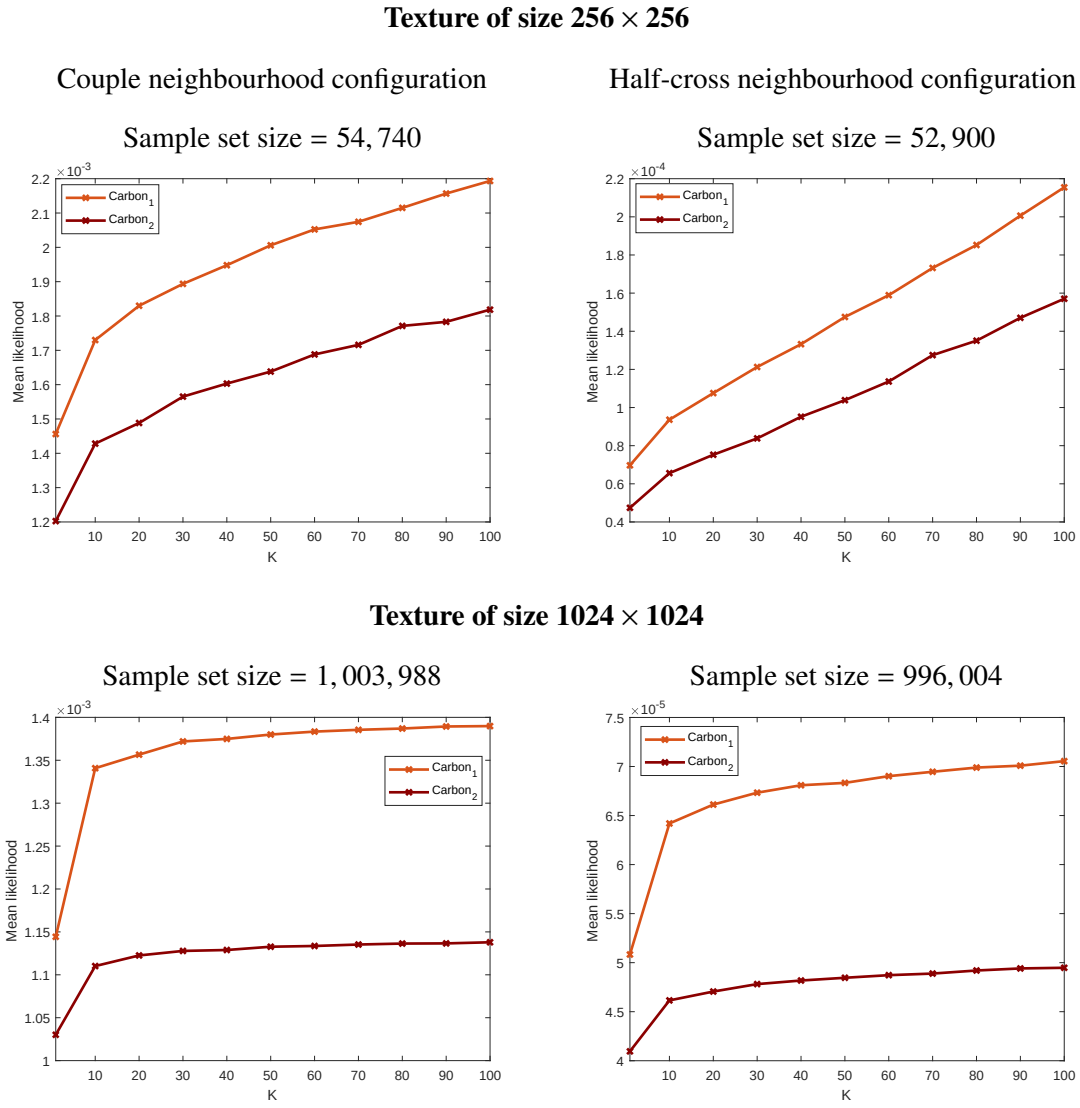


Figure 2.23: Variation of the mean likelihood value over all of a texture's pixels for the LE mixture models with respect to the number of mixture components, K. Two model dimensions are tested: 6 and 9 corresponding to the pair (2-tuple of structure tensors) and half-cross neighbourhoods topologies (3-tuple of structure tensors), respectively. The neighbourhood inter-pixel distance is equal to 8. The variation curves for the Carbon_1 and Carbon_2 256×256 size texture patches are illustrated on the first row and for 1024×1024 texture patches of the same carbon materials, on the second row.

This test scenario illustrates a relatively simple case, when the data forms 3 well-defined groups in the space. However, real data is often more scattered. In addition, the real number of mixture components can be well superior to 3. Thus, in such a case, large sets of data would be necessary so that each group is well represented and statistical inference can be done with no difficulties.

Figure 2.23 illustrates the mean likelihood variation of the LE multivariate Gaussian mixture model, with respect to the number of mixture components, K. Statistical models of dimensions 6 and 9 have been considered. They are employed for characterising distributions of extended structure tensors corresponding to couple and half-cross neighbourhood topologies. The results correspond to the 256×256 large carbon material textures Carbon_1 and Carbon_2. Larger texture samples of

1024×1024 pixels of the same materials have been considered as well. The considered patch sizes result in extended structure tensor sample sets of around $50,000$ and $1,000,000$ samples large. The number of mixture models is varied between 1 and 100 and a step of 10 is considered between consecutive likelihood values. The last row of Figure 2.23 illustrated the variation of the number of degrees of freedom of the 6 and 9 size statistical models, with respect to the number of mixture components, K. In the case when the sample set size is around $50,000$, the mean log-likelihood is not converging towards an optimal number of mixture models. The same behaviour is observed as in the previous case of simulated data, for small sample set sizes (see Figure 2.22). On the contrary, for larger sample sets, a more stable behaviour of the likelihood function is observed for both dimensions of the statistical models (second row of Figure 2.23). In addition, a high sensitivity of the likelihood function to the statistical model's dimension is observed in the case of real data. More precisely, in the case of half-cross neighbourhood configuration, the log likelihood curve has a more unstable character than in the case of couple neighbourhood configuration, and that for sample sets of around $1,000,000$ samples large.

The differences in the aspect of the mean likelihood curves, when dealing with an order 3 increase in the dimension of the statistical model (equivalent to adding one point to the neighbourhood configuration) are more marked for real than for simulated data. This observation might suggest that the optimal number of Gaussians in the mixture model necessary for fitting the data is well superior to 3. Large sample sets are necessary for performing statistical inference on real data of higher dimensional spaces, whose repartition in space can be scattered, as opposed to simulated data whose repartition is space is controlled.

2.4 Dealing with Rotation Invariance

The LST field encompasses the orientation and anisotropy information in each point of a texture. However, to some applications, the orientation is not informative to the texture's description, but the anisotropy alone.

Let us consider a classification application of directional textured images of known class. While each class is characterized by a particular textural pattern, the texture of interest appears in different orientations. Given two textures, their similarity should be assessed in terms of the textural pattern, namely of the anisotropy characteristics and not according to the textures' orientations. More precisely, given the same texture in two different orientations, one would want the distance between the two images to be equal to 0. Thus, in such applications, rotation invariance is a real prerequisite of the texture analysis method employed.

2.4.1 Rotation Invariance for Marginal LST Statistical Models

Two strategies for addressing rotation invariance of marginal LST based methods are proposed in this work. A first strategy relies on determining the global directional tendency of a texture and then

imposing orientation corrections to the statistical model describing the texture's LST field. A second strategy does not rely on an a priori orientation estimation. Instead, it does an exhaustive search of the orientation wise best match between two textures.

Global Orientation Estimation based Rotation Invariance

The success of this approach lies upon an accurate estimation of a texture's dominant orientation, θ. To this purpose, two techniques are proposed:

- the global texture orientation, θ, is given by the orientation of the LST field's center of mass. The center of mass is estimated in terms of the AI or LE metric, according to the LST statistical model further employed for texture characterisation;
- the global texture orientation, θ, is given by the modal value on the histogram of orientations of all the structure tensors in the field.

Once a texture's dominant directional tendency estimated, all the structure tensors in its LST field are applied a rotation with the rotation matrix $\mathbf{R}$ corresponding to the previously estimated angle, θ. Given the texture's original LST field $\mathbf{Y}$, its rotated counterpart $\mathbf{Y}_{rot}$ is obtained as follows:

$$\mathbf{Y}_{i_{rot}} = \mathbf{R}^{\dagger}\mathbf{Y}_i\mathbf{R} \tag{2.12}$$

for $i = 1, ...N$, N being the number of tensors of a texture. Thus, the dominant orientation of the resulting LST field $\mathbf{Y}_{rot}$ will be equal to 0°. The statistical model's parameters are further estimated on the rotated LST field and the approach becomes, thus, rotation invariant.

Exhaustive Search based Rotation Invariance

When comparing two textures, the original LST field of one texture is applied successive rotations by all the t evenly spaced angle values in the pre-defined interval $[\theta_{r_1}, \theta_{r_2}]$, as given in equation 2.12. This results in t rotated LST fields for the first texture while the second texture's LST field is left unchanged. A statistical model will then be estimated for every LST field and a distance will be computed between each of the t statistical models corresponding to the first texture and the statistical model of the second texture. The smallest distance in the set will be considered as the real distance between the two textures. While the exhaustive search leads to a reliable rotation invariance, it takes a considerably longer implementation time compared to the methods based on the estimation of a texture's main orientation.

An example of two anisotropic textures of maritime pine forest where a rotation invariant texture analysis approach is necessary in a classification context is given in Figure 2.24. A quick visual check shows that the textures Pine_3 and Pine_4 have quite similar patterns and they are different mainly in terms of their orientations. These aspects are confirmed by the LST parameter maps:

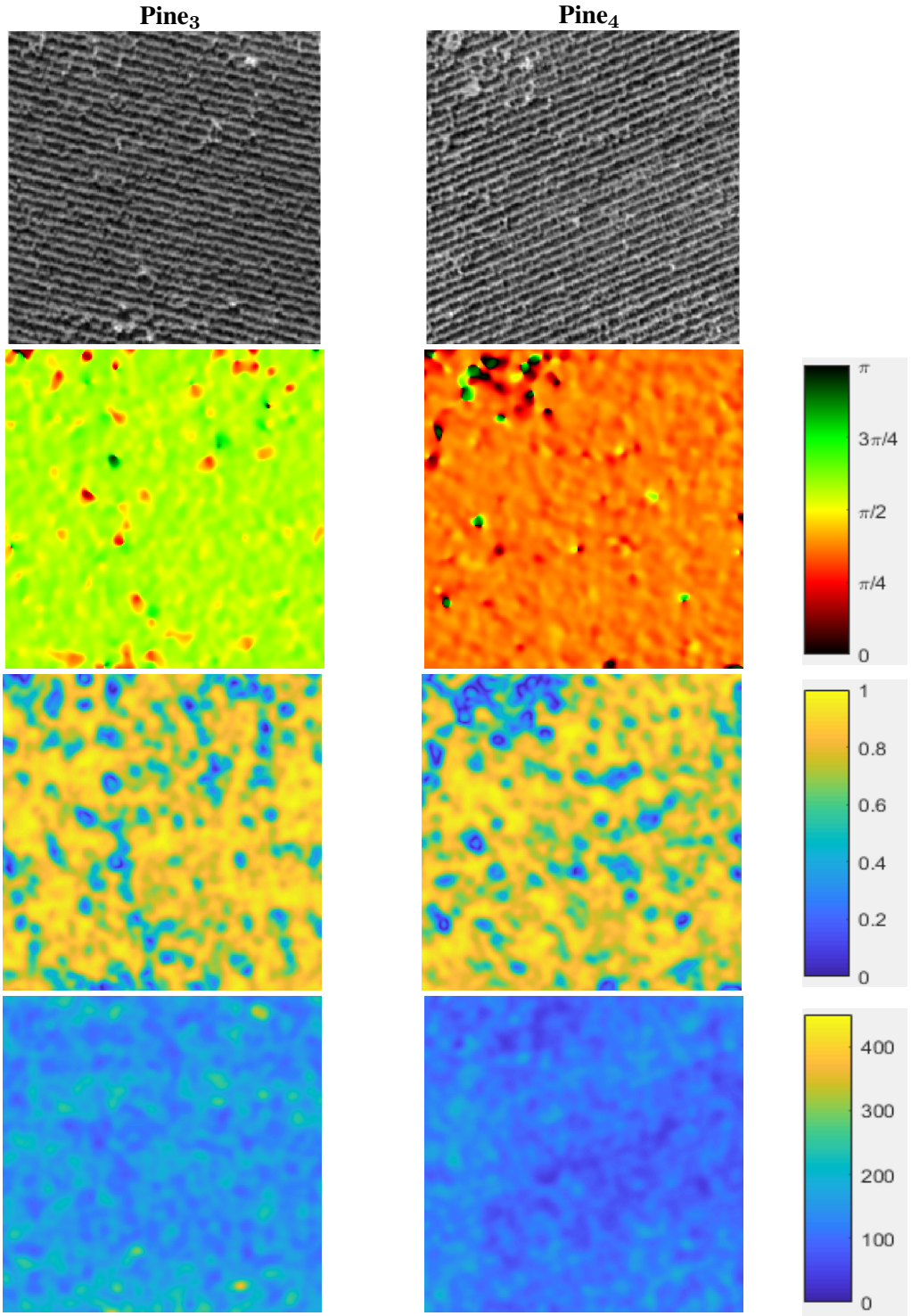


Figure 2.24: LST parameters maps of two maritime pine forest textures before the rotation of the LST fields. The LST fields were calculated for the following values of the parameters: $\sigma_G = 1$ and $\sigma_T = 3$. From top to bottom - original textures, orientation, coherence and energy maps.

similar maps for the coherence and energy parameters and different maps for the parameter θ. The LST fields of the two textures are rotated with the angle corresponding to the orientation of their corresponding centers of mass. The LE center of mass was considered in this example. The LST orientation maps after rotation are given in Figure 2.25. It can be observed that the orientation maps

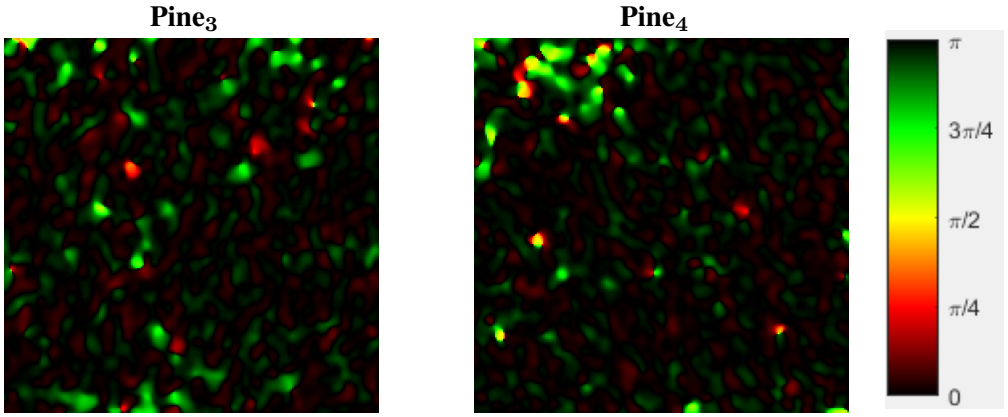


Figure 2.25: LST orientation maps of the maritime pine forest textures Pine_3 and Pine_4 after the rotation of their LST fields with the orientation angle corresponding to their centers of mass.

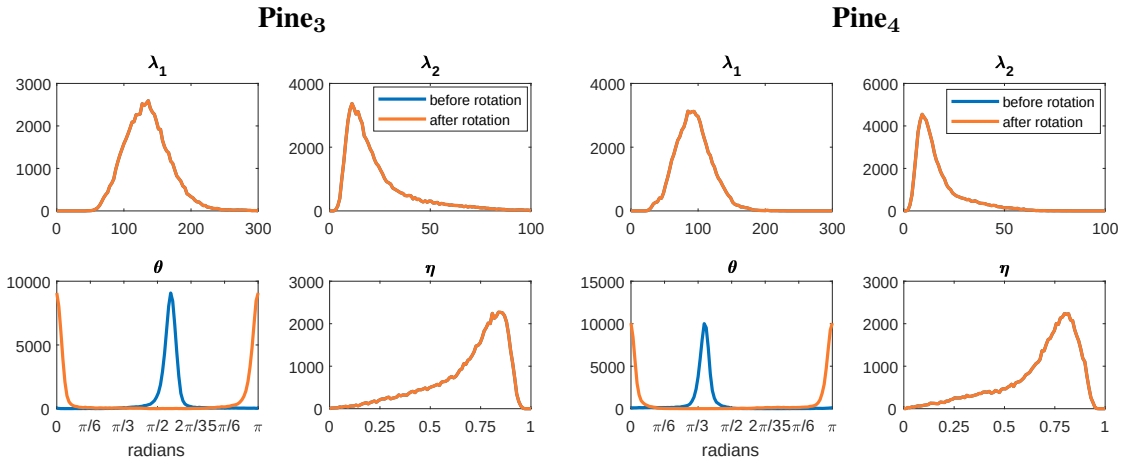


Figure 2.26: The distributions of the LST fields' parameters of the maritime pine forest textures Pine_3 and Pine_4, before and after the rotation of their LST fields with the orientation angle corresponding to their centers of mass.

are globally very similar after rotation. As for the other LST parameters, they are not affected by the rotation, as it can be noticed on their distributions, illustrated in Figure 2.26. Applying a rotation to each tensor of a texture's LST field with a given angle, corresponds to applying a translation to the LST field's orientation distribution with a value equal the orientation of the center of mass, as it can be observed in Figure 2.26. The distribution of the LST fields parameters after rotation are very similar for the two textures Pine_3 and Pine_4.

While a rotation invariant approach is well adapted for strongly anisotropic textures having a dominant global orientation, such an approach is questionable for textures where the notion of global orientation is ambiguous. An oyster park texture is given as example in Figure 2.27. Its pattern is composed of oyster racks displaced upon 3 different orientations, visible on the LST orientation map and on the orientation distribution that has 3 modes. The LST parameters distributions and the orientation map after applying a rotation to all the structure tensors in the field are given as well in Figure 2.27. The orientation of the LE center of mass is considered when applying the

rotation. In this case, the orientation of the center of mass does not well represent the texture's global orientation. Instead, it is the weighted mean of the 3 different orientations corresponding to the oyster racks displacement. As the global orientation of the texture is not well-defined in this case, the rotation invariance strategies based on the estimation of the LST field's global orientation are not well-adapted. For such a case, among the methods proposed here for handling rotation invariance, the exclusive search one is the most adapted.

The methods for dealing with rotation invariance proposed in this work have been assessed in a classification context on different textured images databases, the results being repported in [Noutatiem Guiafaing 2016]. For the exhaustive search method, the angle values are ranged from 1° to 180° with an incremental step equal to 1°, resulting thus in 180 LST fields for each texture. The experiments presented in the report show no sustained overall gain in the classification performances of one approach over the other. In the absence of a clear winner, we choose to apply in the experimental settings further proposed in this work the least complex rotation invariance strategy amongst the different propositions. It is based on estimating a texture's main orientation by the orientation of its LST field's center of mass. The center of mass is estimated by the LE or AI metric, according to the nature of the LST statistical model employed.

2.4.2 Rotation Invariance for Joint LST Statistical Models

When it comes to statistical models for characterising LST neighbourhoods distributions, the rotation invariance approach is slightly different. While the strategy applied for marginal LST distributions still applies, a supplementary step is implemented. Namely, in addition to rotating the individual structure tensors composing a texture's LST field, the pixels' positions defining a given neighbourhood configuration are applied as well a rotation with the orientation angle θ corresponding to the estimated center of mass. In this way, the considered pixel neighbourhood configuration is orthogonal to the textural pattern, as illustrated in Figure 2.28. When the rotated position does not correspond to an actual pixel position (it is not an integer), the nearest neighbouring position is considered.

A particularity of the proposed LST statistical modelling based approaches is that rotation invariance is an integrated part, since the orientation estimation is intrinsic to the local texture descriptor. On the contrary, this is not the case for most of the state of the art texture analysis methods that attain a rather *pseudo* rotation invariance. In these cases, rotation invariance is sometimes addressed by applying additional processing steps to the database, prior to the employment of the texture analysis approach. The pre-treatment consists of estimating the texture's global orientation and then rotating all the textures in the database so that their orientation is equal to 0° [Regniers 2014a]. In some other cases, the rotation invariance is approached by applying different modifications to an original texture analysis method. For example, for the GLCM approach, the texture descriptors are computed for several different orientations and then averaged in order to achieve a rotation invariant texture

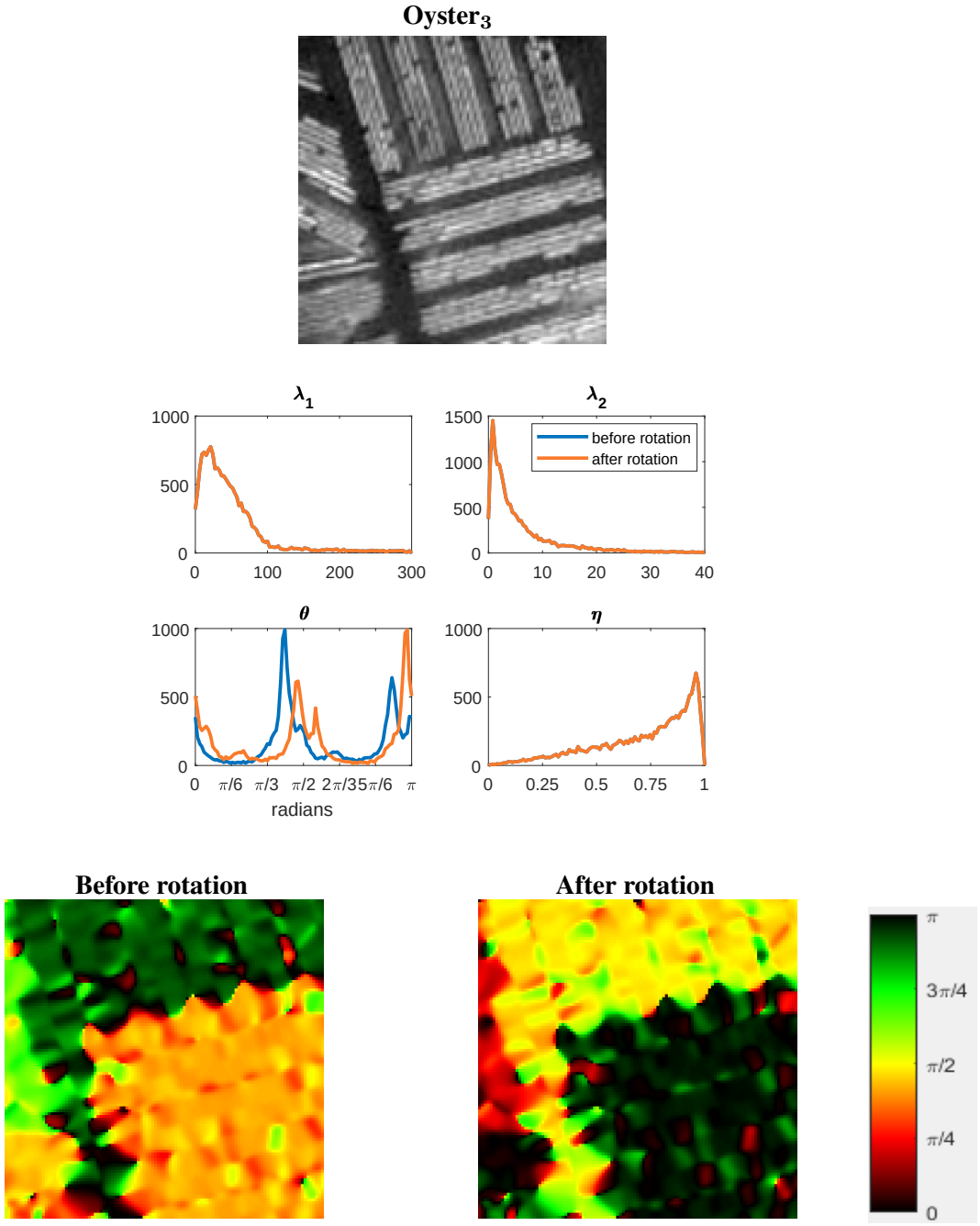


Figure 2.27: The LST field's parameters of an oyster park texture, before and after rotation of the LST field with an angle corresponding to the orientation of the center of mass. The LST field was calculated for the following values pf the parameters:$\sigma_G = 1$ and $\sigma_T = 2$. From top do bottom: original texture, distributions of the parameters of the LST field and orientation maps before and after rotation.

descriptor [Haralick 1979, Maillard 2003]. The drawback common to these solutions is an increased computational time necessary for the additional steps implemented in order to ensure rotation invariance. These steps are extrinsic to the texture analysis method, as opposed to the LST based methods where the rotation invariance is intrinsic to the method.

It should be noted that the different strategies for addressing rotation invariance are valid in the

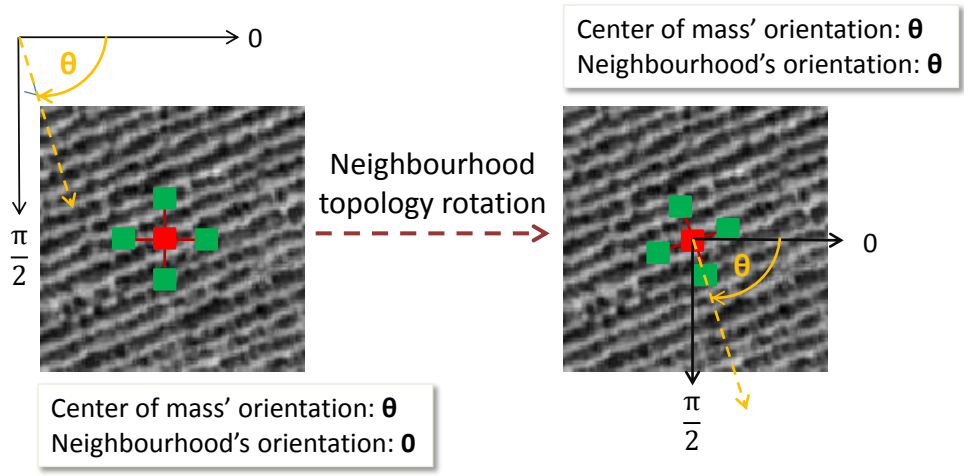


Figure 2.28: Neighbourhood topology rotation for ensuring an RI characterisation of an anisotropic texture's spatial information. Left to right: original and rotated neighbourhood configurations, according to the LST's center of mass orientation, θ. The neighbourhood pixels are represented at a larger scale than the original one, for visualisation reasons.

limit of directional textures presenting a clearly defined dominant orientation. These methods are prone to be biased for textures presenting either isotropic patterns either a mixture of differently oriented anisotropic patterns for which the notion of dominant orientation is rather ambiguous.

2.5 Conclusions

In this chapter, the proposed LST statistical modelling methodology for texture characterisation has been introduced.

The chapter started with an introduction on textured images. The different definitions in addition to the role of texture in the visual perception mechanisms have been given first. Second, the main properties and types of textures have been discussed. The main computer vision application fields for textured images were identified and a brief synthesis of several commonly employed texture analysis methods was further made. They are divided into four categories: structural, descriptive statistics, frequency domain filtering and model based approaches.

The proposed approaches are a combination of structural and model based texture analysis methods. On the one hand, they are structural methods since we employ the local structure tensor tool for locally characterising a texture in terms of its underlying geometry. The local geometry information is extracted in each pixel of an image, which results in a local structure tensor field for the entire texture. On the other hand, the approach is model based since we fit a statistical model to the previously estimated LST field.

The local structure tensor was described in Section 2.2. In addition to its computation, different ways of representation were given, based on eigenvalue decomposition. More precisely, the LST

can be represented by means of the following parameters: its eigenvalues λ_1 and λ_2, its orientation θ and additional parameters computed from the eigenvalues. They consist of the coherence parameter, indicating the texture's degree of local anisotropy and of the energy parameter, an indicator of the local level of contrast. Two representation ways of a texture's LST field are given, by means of the LST parameters maps or by means of ellipse fields. The advantages and disadvantages of each representation approach have been discussed, for several example textures. The adaptability of the structure tensor tool to the scale of interest of a given texture has been further showed. The scale of analysis can easily be adjusted, by tuning the standard deviations of the Gaussian derivative and Gaussian weighting kernels employed when computing the structure tensor.

The proposed LST statistical modelling based approaches for texture characterisation were presented in Section 2.3. A first method consists of describing marginal distributions of structure tensors. The LE multivariate Gaussian model and the Riemannian Gaussian model defined on the AI metric space have been considered to this purpose, in addition to their corresponding mixture models. The fit of the proposed statistical models to empirical LST distributions of various textures has been analysed. While one Gaussian (AI or LE) is enough for characterising relatively simple texture patterns, mixture models are more adapted when the texture is composed of inter-winding patterns of different characteristics. Naturally, the order of the mixture model increases with the increase in the complexity of the textural pattern. For a same order of the statistical model, LE models showed a better fit to the empirical LST field than the AI models. In some cases, one LE Gaussian model showed a better fit than a mixture of Gaussian models on the AI space. In addition, the LE approaches are significantly faster than their AI counterparts.

Methods for characterising distributions of p-tuples of structure tensors have further been developed on the LE metric space, to the purpose of characterising spatial and multiscale dependencies in LST fields. Multivariate Gaussian and Gaussian mixture models of higher dimensions are considered for characterising p-tuples structure tensor distributions. While extracting the data encompassing the spatial and multiscale information is an easy task on the LE metric space, increasing the dimension of the observation space has shown to induce instability to the statistical inference process.

In the last part of this chapter, different strategies are proposed for ensuring the rotation invariance of the LST statistical modelling methods for texture analysis. The proposed strategies follow two directions: global orientation methods and exclusive search methods where the best orientation wise match is searched between the statistical models describing two textures. While the first methods give satisfying results when dealing with anisotropic textures presenting one dominant orientation, they are not adapted for textures where the global orientation is ambiguous. The exclusive search method suits best these cases, out of the proposed strategies for dealing with rotation invariance.

Chapter 3

Recognising Texture with Models of LST Fields

"The first thing the intellect does with an object is to class it along with something else. But any object that is infinitely important to us and awakens our devotion feels to us also as if it must be sui generis and unique. Probably a crab would be filled with a sense of personal outrage if it could hear us class it without ado or apology as a crustacean, and thus dispose of it. 'I am no such thing,' it would say; 'I am MYSELF, MYSELF alone.'"

–William James, American Philosopher

Contents

3.1 Introduction and Objectives

The main objectives of this chapter consist in assessing the capabilities of all the proposed LST statistical models of characterising the LST field of textured data and of further recognizing and discriminating these textures. Several experimental settings are considered relative to the classification of different types of textured images. The applications are content based image retrieval (CBIR) and supervised classification. They concern two types of textured images, namely very high resolution (VHR) remote sensing images and carbonaceous material images issued from high resolution transmission electron microscopy (HRTEM) technology.

A comparison between the efficiency of the statistical models on the two metric spaces, i.e. affine-invariant and log-Euclidean, is made. In addition, for each metric, the differences between their corresponding statistical models are analysed. On the log-Euclidean metric space, a framework for modelling the spatial dependencies of an LST field is assessed as well, in addition to the statistical models for describing marginal distributions of structure tensors.

All of the proposed methods are further compared with different state of the art texture analysis methods. The comparison between the different methods is made in terms of classification accuracy, computation time and rotation invariance capabilities as well, the later being an actual prerequisite of the texture characterisation methods when dealing with anisotropic texture classification applications.

3.2 Test Data

The following paragraphs focus on the datasets used for experimentally validating the LST statistical models for characterising the LST field of a textured image. They consist of real-world data and concern remote sensing and material imagery.

3.2.1 Oyster Field Textured Patches

The first dataset used in our experiments is composed of VHR remote sensing images of oyster parks texture patches, that are grouped in 3 classes: cultivated oyster racks, abandoned oyster fields and foreshore.

There is a high pressure around the oyster production due to some difficulties arising from different factors. They can be human factors such as some conflict over the land use of coastal areas or ecological factors such as the increasing level of ocean water pollution, the development of some toxic plankton species or the spread of different diseases that the oysters are prone to [Regniers 2014a]. These shortcomings often result in cultivated oyster racks abandonment. These fields remain interesting for oyster production since they represent potential areas for future farming. Cleaning abandoned racks and preparing them for cultivation implies some costs that increase with the time they have been abandoned for - hence the need of identifying and managing these areas

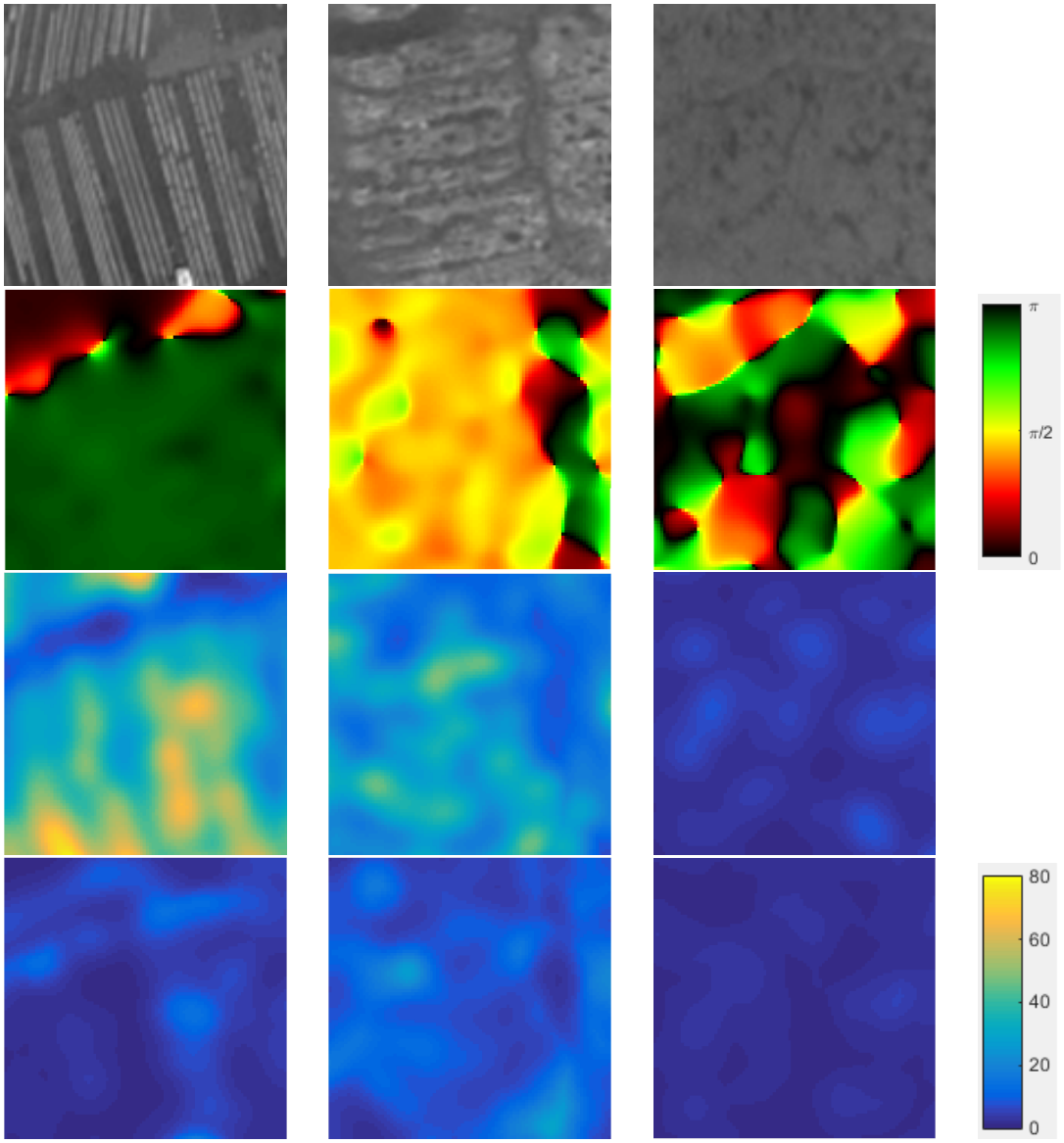


Figure 3.1: Examples of oyster field textured patches (first row) for classes C_1 to C_3 (from left to right) and their corresponding LST parameters fields (from second to fourth row): the orientation θ and the eigenvalues λ_1 and λ_2. The color palettes are on the last column. The LST is computed with the following values for the parameters: $\sigma_G = 1$, $\sigma_T = 5$.

within the shortest delay. Remote sensed images are increasingly used for the automatic management of oyster fields [Choe 2012, Regniers 2014c, Le Bris 2016, Pham 2016b]. On the one hand, they provide information relative to land cover while on the other hand, they represent useful means for quantifying the oyster production over a given area. Thus, there is a growing need for developing different strategies and methods for characterising the content of these images.

The oyster parks patches database was built by extracting 128×128 pixels homogeneous patches from the panchromatic band of a PLEIADES image of spatial resoultion 0.5 m acquired at low tide on the 25th of April 2013. The satellite image covers a large central part of Arcachon Bay, a mesotidal lagoon of 180 km^2 located on the Atlantic Coast in the South of France [Regniers 2014c].

The database is composed of 320 patches, divided between the 3 different classes as follows: 171 patches belong to the class of cultivated racks C_1, 35 are part of the abandoned fields class C_2 and 114 patches belong to the last class C_3 of foreshore.

Figure 3.1 shows texture patch examples belonging to each oyster parks class with their corresponding LST parameters field, namely the orientation and the two eigenvalues. The first class is composed of cultivated oyster racks of length varying from several tens to a hundred meters. Several racks are generally grouped together (2 to 5) at a distance of 1 to 2 meters [Regniers 2014a]. The groups of racks can be displaced at different distances and orientations relative to one another. In addition, some of the oyster racks might be empty, all these inducing, thus, a strong intra-class and even inter-patch diversity (see Figure A.1 of Appendix A).

In terms of texture information, it can be noticed (see Figure 3.1) that the first two classes are composed of structured anisotropic patterns while the last class of foreshore is characterised by a mostly isotropic textured pattern. This class is also more homogeneous. Some more examples of texture patches belonging to each class are given in the A.1 of Appendix A in order to illustrate the intra-class diversity that is certainly challenging for the classification.

3.2.2 Maritime Pine Forest Stands

The second VHR remote sensing image collection used in this study in order to validate the capacities of LST statistical models for texture characterisation consists of maritime pine forest stands. They present different shapes and sizes and are grouped in classes according to their age.

Cultivated forest management follows a series of techniques and land operations that are part of a well-defined protocol [Regniers 2014a]. Thus, it can be seen as a cyclic process of well-known phases. This cyclicity enables to make correlations between the underlying forest structure variables such as: age, crown diameter, trees height, etc. and the spatial distribution of the trees. Thus, VHR remote sensing images can be seen as a useful tool for characterising and discriminating between different forest populations, by means of texture analysis methods exploiting the spatial pixel organisation. These techniques facilitate the automated generation of forest structure variables maps - forest inventory tools for biomass estimation or prediction of future production. The increased interest in remote sensing textured image analysis as a tool for forest land management is shown also by the numerous research papers addressing this topic [Kayitakire 2006, Beguet 2012, Champion 2014, Regniers 2014b].

The study site is the *Landes de Gascogne* forest, situated in the South West of France. It covers 100 ha of cultivated maritime pines, organized in uniform age stands ranging from 0 (clear cuts) up to approximately 50 years old. The image database consists of maritime pine forest stands of various shapes and sizes extracted from a PLEIADES image of spatial resolution 0.5 m, acquired on the 8th of August 2012. Ground-truth is available, consisting of a priori knowledge of the contour and class of every stand. The ground-truth consists of 179 forest stands divided between the different classes

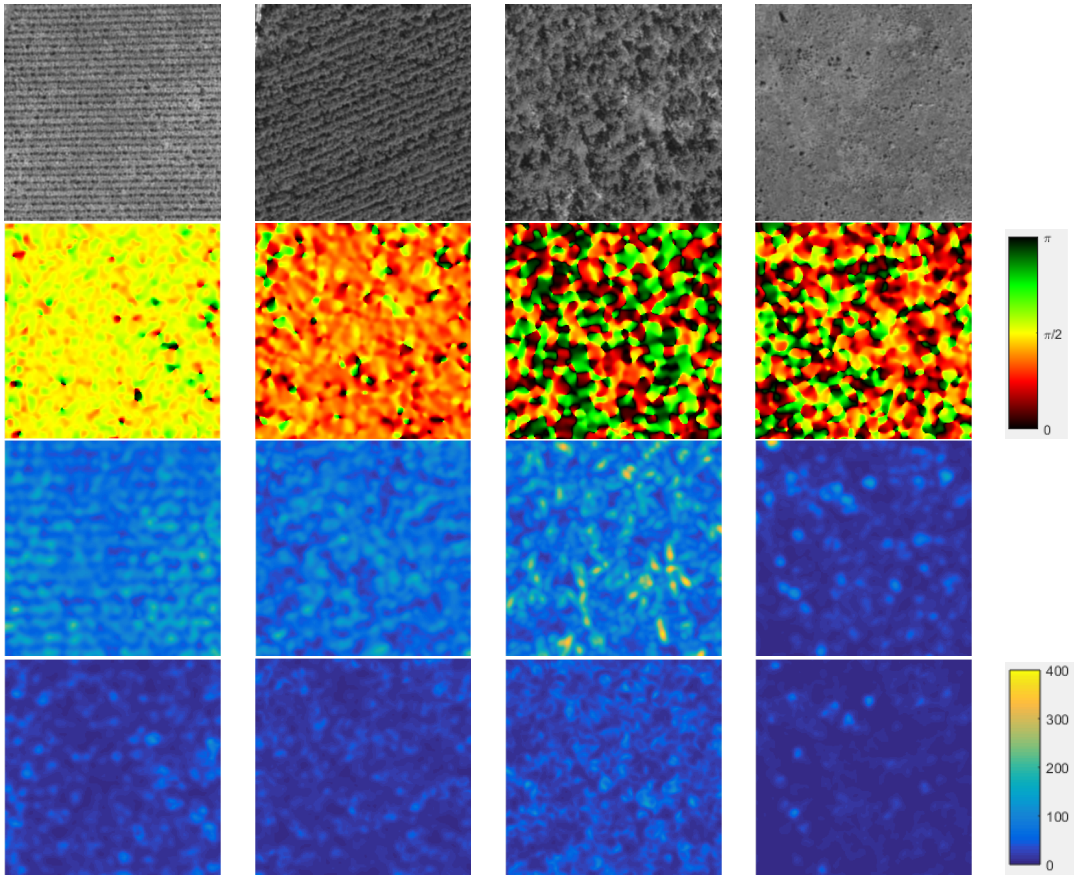


Figure 3.2: Examples of maritime pine forest textured patches (first row) for classes C_1 to C_4 (from left to right) and their corresponding LST parameters fields (from second to fourth row): the orientation θ and the eigenvalues λ_1 and λ_2. The color palettes are on the last column. The LST is computed with the following values for the parameters: $\sigma_G = 1.9$, $\sigma_T = 3$.

as follows: 67 belong to C_1 ($0-9$ years old), 57 to C_2 ($10-19$ years old), 40 to C_3 (>20 years old) and 15 to C_4. Example patches of size 256×256 extracted from forest stands of every age class and their respective LST parameters fields are given in Figure 3.2.

The forest evolution over time (larger tree crowns, wider canopy, lower tree density) translates into changes in the spatial organisation of the textural pattern. Not only the size of the objects becomes larger, but the structure becomes less organised meaning that the texture becomes more isotropic with the ageing of the forest population. Variations in the texture structure might appear due to some other factors as well, such as storms or other natural phenomena that might significantly damage the forest [Regniers 2014a].

In the following we briefly present the particularities characterising each forest class. For the first age class ($0-9$ years old) the tree ranks as well as the inter-rank forest floor and understory are visible on the image, due to a small diameter of the crown size (1 m). This class is thus characterised by a regular and anisotropic textural pattern that shows on the LST parameters maps as a higher level of uniformity in the local orientation estimates and higher differences in the range values of λ_1 and λ_2 - see Figure 3.2.

For the second forest class, the tree ranks are still visible, although the crown diameter increases, ranging from 1 to 3 m, resulting in a wider canopy. Thus, the textural content of this class is still anisotropic - see Figure 3.2.

When it comes to class 3, the tree ranks are no longer visible on the image as tree crowns are wider than 3 m. There is also more variety in the population density and vegetation holes might appear in some areas due to thinning operations and natural phenomena [Regniers 2014a]. The textural pattern characterising this class looses, in this way, its organised and anisotropic properties. This appears on the LST parameter maps as a mixture of different values for the local orientation and a lower discrepancy between the range of values of the two eigenvalues - see Figure 3.2.

Finally, the fourth class of clear cuts consists of images of the soil after the removal of all trees. However, some other small-size vegetation might be visible on this areas. The textures observed in this class are mostly anisotropic - see Figure 3.2.

3.2.3 Carbon Composite Material Textured Patches

The third database used for experimentally validating the texture characterisation capabilities of the proposed LST statistical models is composed of carbonaceous material lattice fringe (LF) images issued from high resolution transmission electron microscopy. They are grouped in directional texture classes with a higher degree of homogeneity than the textures in the VHR databases.

HRTEM is a useful technique for investigating the nanostructure of different carbonaceaous materials. The effective characterisation of the spatial arrangement of fringes enables to describe, identify and distinguish between different types of materials. Since LF imaging visual analysis is prone to human errors and is time consuming, the development of automatic image analysis methods is a need addressed in many scientific papers [Shim 2000, Rouzaud 2002, Germain 2003, Toth 2013, Da Costa 2015].

The database consists of texture patches extracted from carbon material snapshots issued from high resolution transmission electronic microscopy. The patches are grouped in 4 classes (see Figure 3.3): C_1 - PAN based carbon fiber, C_2 - regenerated laminar PyC (pyrocarbon), C_3 - rough laminar PyC and C_4 - smooth laminar PyC (for more details about the materials see [Da Costa 2015]). Each class contains 16 patches of size 256×256 extracted from HRTEM snapshots of size 2048×2048 representing the different material nanostructures (see Figure B.1 in Appendix B).

The spatial arrangements of fringes form a textural pattern on the HRTEM images that is specific to each material. The image of the first class of carbon fiber is composed of layers of folded fringes disposed in groups. Thus, at a large scale of analysis, it is characterised by an isotropic texture as there is no global dominant orientation. This spatial organisation is more noticeable on the original HRTEM snapshot (see image in Subfigure B.1a of appendix B) than on the texture patches.

For the second and third classes of rough and regenerated laminar pyrocarbons, the fringes are arranges on longitudinal layers, which gives a strong anisotropic textural character. The anisotropy

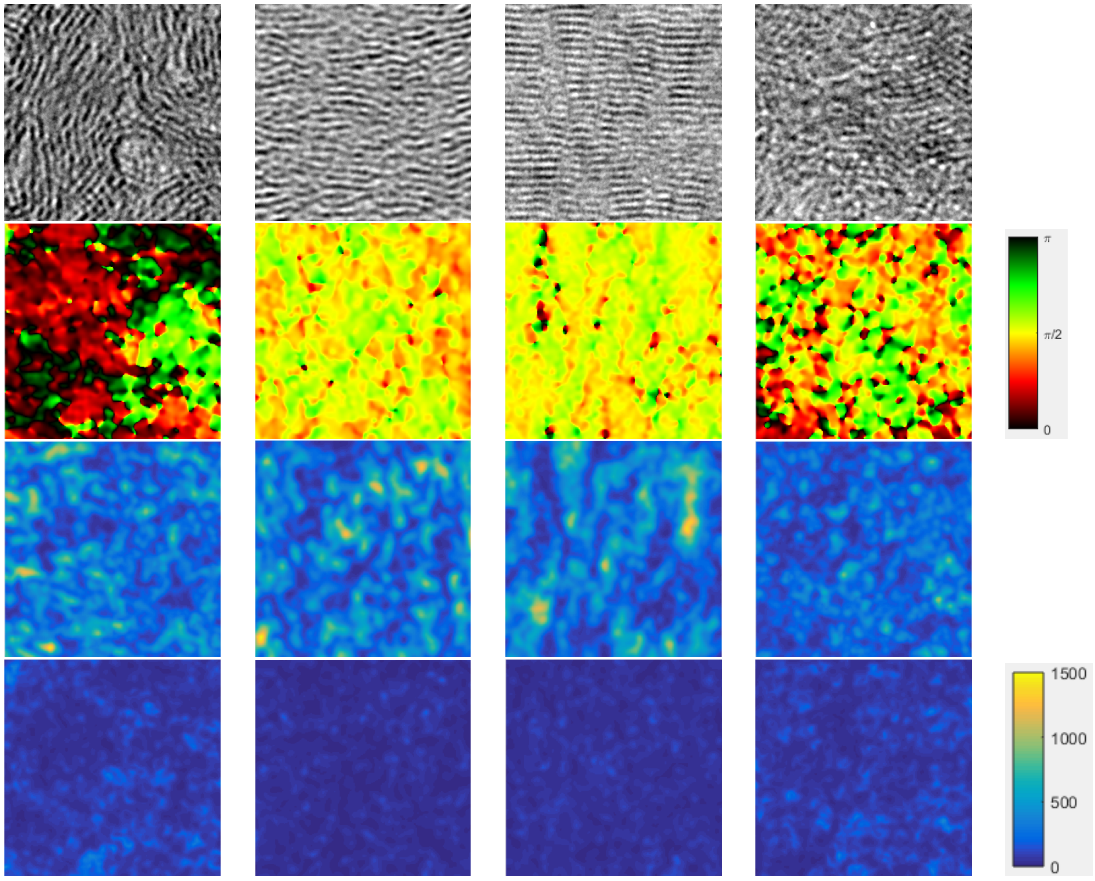


Figure 3.3: Examples of HRTEM carbonaceous material textured patches (first row) for classes C_1 to C_4 (from left to right) and their corresponding LST parameters fields (from second to fourth row): the orientation θ and the eigenvalues λ_1 and λ_2. The color palettes are on the last column. The LST is computed with the following values for the parameters: $\sigma_G = 1$, $\sigma_T = 3$.

shows as a high level of homogeneity on the orientation map, as estimated by the LST (see Figure 3.3). The textural patterns of these two materials are very similar, one of the slight differences being the fact that the regenerated PyC features a greater deal of structural defects [Da Costa 2015].

As far as the last class of smooth laminar PyC is concerned, the textural pattern is still anisotropic but to a lower extent compared to the two previous classes, as there is a higher amount of local variations. This aspect can be noticed as well on the LST orientation map, manifested as a mixture of different local orientations (see Figure 3.3).

As a global remark, this database is characterized by a higher level of intra-class homogeneity, as opposed to the other datasets considered in this work.

In order to test the invariance of the proposed methods to the texture's main orientation in an image retrieval context, each patch of a class is applied a different rotation with an angle randomly chosen in the interval $[0°, 180°]$. In addition, all images in the database are imposed the same dynamic range by standardization to overcome differences due to the manual adjustments of the microscope.

3.2.4 Land Use Land Cover Merced University Database

The UC Merced LULC database was formed by extracting homogeneous patches from USGS (United States Geological Survey) National Maps covering different US regions.

As opposed to the previous datasets that are mainly composed of directional textures, LULC database is far more heterogeneous. It consists of both textured and non-textured LULC orthorectified aerial images grouped in 21 classes: agricultural, airplane, baseball diamond, beach, buildings, chaparral, dense residential, forest, freeway, golf course, harbor, intersection, medium residential, mobile home park, overpass, parking lot, river, runway, sparse residential, storage tank, tennis court. Each class contains 100 images of size 256×256 and pixel resolution of 30 cm [Yang 2013]. It represents the largest dataset of this kind publicly made available. Given the large number of samples, this dataset is more adapted to CBIR applications, compared to the other datasets used for this purpose in this work. However, given that it is composed of non-textured image patches as well, it is less adapted for testing the capabilities of the proposed LST statistical models as these approaches are texture specific. However, it has been considered as an interesting choice to test the methods on this dataset as well, in order to have access to immediate comparison to a broader set of state of the art texture and non-textured remote sensing image analysis methods [Yang 2013, Aptoula 2014, Özkan 2014]. This gives, thus, the possibility to place the LST statistical modelling methods for texture analysis on the wider map of remote sensing image analysis approaches.

This database is composed of RGB images. However, in our experiments, only the gray level information has been exploited. This dataset is characterised by a significantly pronounced class and inter-class diversity. Example patches of each class are given in the Figure A.3 of Appendix A.

3.3 Texture Recognition Protocols

We are considering two main image classification methods for assessing the capabilities of the proposed statistical models for characterising the LST field of a textured data. They consist of a content based image retrieval protocol for browsing textured image patches and a supervised classification protocol for grouping textured image regions into different classes. The main concepts, performance indicators as well as the algorithms and the classification protocols relative to each classification method are briefly described in the following paragraphs.

3.3.1 CBIR

Content based image retrieval is an image browsing technique that relies on the existence of an image or image patches database, where each image or image patch belongs to a pre-defined known class. Its objective is the automatic retrieval of images from the database, according to different features characterising the content of an image, such as texture, color, shape, etc. In this work the image content of interest is texture. Texture information is captured by statistical models fitted to

the texture's LST distribution or by other state of the art texture analysis methods.

In this work we apply the CBIR algorithm on collections of image patches. A patch is a smaller size image of fixed size extracted from a larger image. It contains a homogeneous textural pattern representative for the class to which it belongs.

The architecture of a CBIR application is illustrated in Figure 3.4. The method is composed of 2 steps. Firstly, a signature is extracted for each image patch in the database. It corresponds to a set of parameters or image features, specific to the employed texture analysis method. Secondly, a similarity measurement is performed. More precisely, for each patch, a distance is computed between its signature and the signature of every other patch in the database. The closest images can be retrieved as a result [Do 2002].

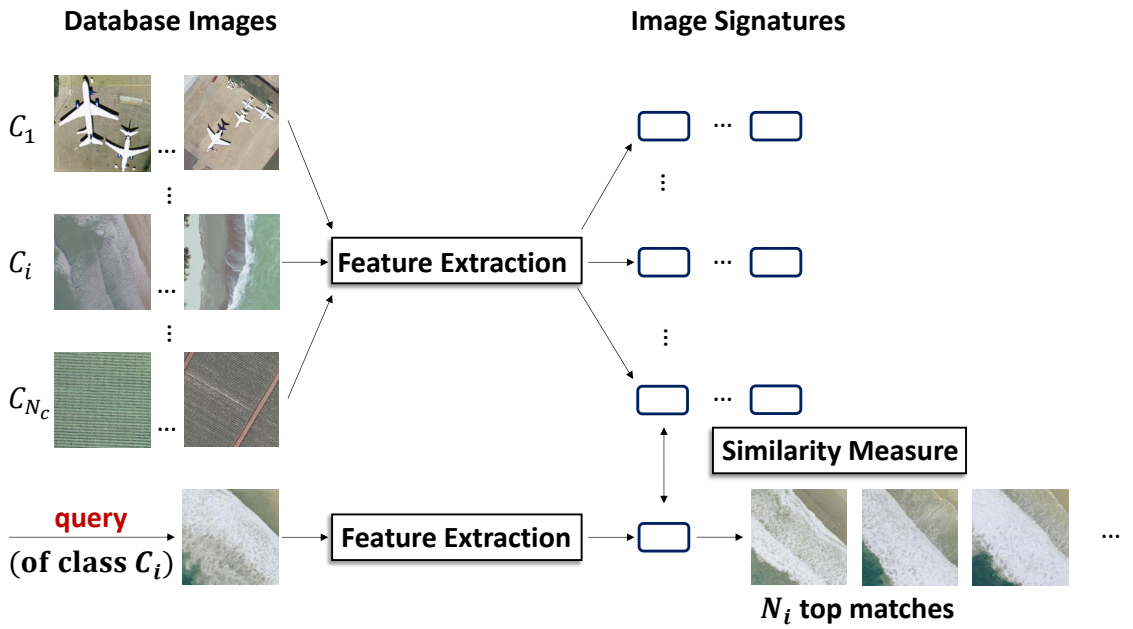


Figure 3.4: Ground truth content based image retrieval architecture - with search in the N_i top matches, where N_i denotes the number of ground-truth patches of same class C_i as the query.

3.3.1.1 Performance Evaluation

Various criteria can be used to evaluate the CBIR performances. In the following we will present two of the most widely used indicators.

The average retrieval rate (ARR) is commonly employed for assessing the results of a CBIR method [Müller 2001]. It gives the average percentage of images of the same class as the query, among the first N_i matches, where N_i is the number of images in class C_i, $i = 1, ...N_C$, N_C being the

number of classes in the database. It is obtained as follows:

$$ARR = \frac{1}{N_T} \sum_{i=1}^{N_C} \sum_{q \in C_i} \frac{n_q(N_i)}{N_i} \tag{3.1}$$

where N_T denotes the total number of patches in the database and $n_q(N_i)$ stands for the number of patches of same class C_i as the query q, in the top N_i matches.

It should be noted that the closest image of the query has the same importance as the image of rank N_i. Consequently, this criterion is sensitive to the presence of outliers in the database.

Some of the databases used in our experiments hold a class size disparity. As a consequence, the retrieval results might be biased, the classes being better represented having a stronger influence on the global result. There are different ways to address this issue. We are presenting in the following two of the proposed solutions, that have been adopted as well in this work in order to facilitate the comparison to state of the art texture analysis methods. These methods consist of statistical modelling approaches of wavelet subband coefficients [Regniers 2013, Regniers 2014a], specific implementations of LBP [Aptoula 2014], Gabor filtering [Yang 2013] or the multiscale texture analysis approach proposed by Aptoula in [Aptoula 2014].

Dealing with class size disparity

The first solution has been proposed by Regniers in [Regniers 2014a] and adopted as well in this work in order to facilitate the comparison to wavelet based texture analysis methods [Regniers 2013, Regniers 2014a]. It consists of a slight adjustment to the standard retrieval protocol. A new database is formed, containing 25 patches per class, randomly selected from the original database. The retrieval performances are then computed on the new database and this procedure is repeated 100 times. The mean ARR and its standard deviation are performed in the end, for all iterations.

The second solution consists of the computation of another performance indicator, namely of a modified form of the ARR criterion. It consist of the average normalized modified retrieval rate (ANMRR) widely used in the MPEG-7 experiments [Manjunath 2001] and in some VHR remote sensing image indexing studies [Aptoula 2014, Özkan 2014, Yang 2013].

It has been proposed in order to address some of the limitations of the ARR criterion, including class size sensitivity. Secondly, imposing the number of relevant matches to N_i is considered to be a hard limiting measure [Manjunath 2001] since a ground-truth patch ranked as $N_i + 1$ does not contribute to the retrieval accuracy while it might still be relevant in terms of subjective retrieval performance. As a remark, in the context of CBIR, the ground-truth consists of the patches belonging to the same class as the query q.

Given a query q of class C_i, each ground-truth patch is assigned a rank equal to the position where it has been retrieved. For the ANMRR computation, the rank is modified, so that the ground-

truth patches retrieved at a position considered too far away for being relevant to retrieval are assigned a constant penalty. This is done by establishing for each class a number of relevant ranks $K_i > N_i$, where N_i is the size of the class C_i. A commonly used value for this parameter in the case of relatively large classes ($N_i > 20$) is $K_i = 2N_i$ [Manjunath 2001, Yang 2013]. The modified rank holds:

$$R^*(k) = \begin{cases} R(k), R(k) < K_i \\ 1.25K_i, R(k) > K_i. \end{cases}$$

The average rank (AR) is computed for each query using the values of the modified ranks:

$$AR = \frac{1}{N_i} \sum_{k=1}^{N_i} R^*(k). \tag{3.2}$$

The modified normalized retrieval rate is further computed as follows:

$$NMRR(q) = \frac{AR(q) - 0.5(1 + N_i)}{1.25K_i - 0.5(1 + N_i)}. \tag{3.3}$$

Its value ranges from 0 (perfect retrieval) to 1 (nothing retrieved) and no longer depends on the value of N_i. Finally, the average modified retrieval rate is computed for the whole database, as:

$$ANMRR = \frac{1}{N_T} \sum_{i=1}^{N_C} \sum_{q \in C_i} NMRR(q). \tag{3.4}$$

3.3.2 Supervised Classification

In order to further assess the ability of the proposed statistical models to accurately describe the LST field of a textured data, a supervised classification scenario is considered as well, in addition to the CBIR experiments.

Supervised or *human-guided* classification algorithms are composed of two main steps. A learning step where the textural attributes are extracted from a collection of reference data of known class, called also training data. The second step consists of the classification itself where the data to be classified is assigned a class according to the features extracted in the training phase and to the decision rules of the employed classifier. Therefore, the classifiers requires to dispose of two datasets. On the one hand, a learning dataset is needed and, on the other hand, a validation data set is necessary for assessing the classification performances.

3.3.2.1 Supervised Classification Algorithms

Two supervised classification algorithms are employed in this work, namely k Nearest Neighbours (k-NN) classifier and Support Vector Machine (SVM). Both algorithms are briefly described in the following paragraphs.

k-nearest neighbours (k-NN)

One of the supervised classification algorithm employed in this work is the k-NN (k - Nearest Neighbours). According to this algorithm, a sample from the validation set is assigned the majority class of its k closest samples from the training set. The distances computation is specific to the the texture analysis method employed, as described in Sections 1.4.5 and 1.3.5 for the LST statistical modelling based approaches.

The choice of k is crucial to classification. While a classifier with a small value for k might benefit a higher classification accuracy (it is more probable that the first top matches belong to the same class as the sample to be classified), it has an increased sensitivity to outliers. This sensitivity can be lowered by increasing the value of k. On the other hand, this might lead to a decrease in performance, since it is more probable to find samples belonging to different classes than the actual class of the sample data to be classified. Thus, when tuning this parameter, a trade-off between accuracy and sensitivity to outliers should be made. In this work, the value of this parameter is fixed to 5, as done previously by Regniers in [Regniers 2014a]. This value is mainly imposed by the size of the lowest represented classes.

Given that originally, all the k closest samples have the same importance, a weighting is applied here so that the distance hierarchy is respected in the decision process, as previously proposed in [Regniers 2014a]. Thus, the ith closest neighbour is given the weight, $k-i+1$, for $i = 1, ...k$. A score is computed for every class by simply summing the weights of the k closest samples that belong to the respective class. Finally, the validation sample is assigned to the class having the greatest score.

SVM

We have secondly employed a more complex classifier, namely the support vector machine (SVM) [Vapnik 1979, Boser 1992]. It is largely used in the remote sensing image classification literature [Yang 2013, Regniers 2016], particularly due to its ability to perform well even when limited data is available - a frequent issue in these applications [Mountrakis 2011].

The SVM belongs to the category of maximum margin classifiers. The idea behind the SVM is the search of an optimal hyperplane separating any two classes by maximising the margin, namely the distance between the hyperplane and its closest data point.

The algorithm is implemented in MATLAB using the LIBSVM library [Chang 2011]. It consists of two steps, namely the training and the classification step itself. Multi-class classification (the case when the number of classes $N_C > 2$) is addressed from a binary *one versus one* perspective. An optimal hyperplane is searched - separating any two couples of classes, leading thus to $N_C*(N_C-1)/2$ decision boundaries.

In most of the cases, the data is not linearly separable on its original space $\mathbb{R}^N$. This issue can be addressed by projecting the data into a higher dimensional space $\mathbb{R}^M, M > N$ where the data becomes linearly separable. The optimal hyperplane is searched in $\mathbb{R}^M$ and then projected back to the original space. However, projecting the input data into a higher dimensional space might arise

computational and memory problems, in the case when the dimensionality of the projection space grows very fast with respect to the one of the original space. This issue is addressed in practice via a method referred to in the literature as the *kernel trick* [Scholkopf 2001]. During the training phase, the hyperplane search is approached as a convex optimization problem. At this point, the training data is uniquely used for computing pairwise dot products. As a consequence, the explicit projection of the data on the transformed space $\mathbb{R}^M$ is not necessary, but only the pairwise dot products results on $\mathbb{R}^M$ [Fauvel 2007]. This can be achieved by using the mathematical properties of kernel functions. Namely, given a couple of data in $\mathbb{R}^N$, a kernel implicitly computes their dot product in $\mathbb{R}^M$ without the explicit transformation of the data points to $\mathbb{R}^N$. As a consequence, non-linear hyperplanes can be learned by the SVM by simply replacing distances between data point couples by the kernel function that gives a measure of their mutual influence.

Various choices are available for the kernel definition. A Gaussian kernel (typically called radial basis function in the machine learning literature) of standard deviation σ is employed in this work.

$$K(s_1, s_2) = \exp\Big\{ - \frac{distance(s_1, s_2)^2}{2\sigma^2} \Big\}, \tag{3.5}$$

where (s_1, s_2) denotes a couple of textural signatures. The distances between the image data textural signatures is specific to the texture analysis method employed, see Sections 1.4.5 and 1.3.5 for the approaches based on the LST statistical modelling and 3.4 for the state of the art texture analysis methods used for comparison in this work.

Secondly, the classification step is performed. A degree of belonging to a given class is further computed according to the distances of the data sample to the $N_C - 1$ hyperplanes separating the respective class from the other classes. These scores are added for each class and the test data sample is assigned the class maximizing this score. Similarly to the training phase of the algorithm, the decision phase is expressed as well in terms of dot product of data points (see detailed equations in [Fauvel 2007]). Consequently, the distances between data points are replaced by the kernel function, an explicit projection of the data points into the higher dimensional space being unnecessary.

3.3.2.2 Performance Evaluation

Different criteria can be evaluated in order to assess the performances of a classification method of remote sensing data [Congalton 1991].

They are computed starting from the confusion matrix, constructed in this study as follows: the lines represent the real classes of the regions (ground-truth instances) while the columns contain the instances in different classes, as predicted by the classifier. Thus, $n(i, j)$ stands for the number of instances of class i assigned to the class j by the classifier.

Firstly, the global classification performance is evaluated. A commonly employed index for assessing the global probability of correct classification is the *Overall Accuracy* (OA):

$$OA = \frac{1}{N_T} \sum_{i=1}^{N_C} n(i,i), \tag{3.6}$$

where N_T denotes the total number of regions to be classified, N_C is the number of classes and $n(i,i)$ denotes the confusion matrix entry on the (i,i) position - the number of ground-truth elements in class i predicted by the classifier as belonging to the class i.

The *Kappa* index can be computed as well [Cohen 1960]. It is a global indicator of the proportion of correct classifications that did not occur by chance:

$$\kappa = \frac{OA - Pe}{1 - Pe}, \tag{3.7}$$

where P_e is the proportion of data expected to be correctly classified by chance:

$$Pe = \frac{1}{N^2} \sum_{i=1}^{N_C} n(.,i)n(i,.), \tag{3.8}$$

$n(.,i)$ being the sum of the elements on the ith column of the confusion matrix and $n(i,.)$ denoting the sum of the entries of the ith line of the matrix. It takes values less than or equal to 1, with a higher value indicating a smaller incidence of correctly classified items due to chance. In rare cases it can be less than 1, indicating weaker classification performances than those expected if the data would be randomly split between the different classes.

Secondly, the classification performances for each class can be evaluated. A first statistic is the *Producer's Accuracy* (PA) that describes the probability of a region belonging to a given class according to the ground-truth to be correctly classified. For the class i, $i = 1, ...N_C$, it is computed as follows:

$$PA_i = \frac{n(i,i)}{\sum_{j=1}^{N_C} n(i,j)}. \tag{3.9}$$

Another statistic for evaluating the classification results for each class, is the *User's Accuracy* (UA). It represents the probability of a region assigned to a certain class by the classification algorithm, to actually belong to that class, according to the ground-truth. The UA for the class j, $j = 1, ...N_C$ is given by:

$$UA_j = \frac{n(j,j)}{\sum_{i=1}^{N_C} n(i,j)}. \tag{3.10}$$

3.3.2.3 Protocol

The first step of the supervised classification protocol consists of extracting the textural signatures from all the reference data. Secondly, a cross-validation approach is considered as proposed in [Regniers 2015a] in order to quantify the potentials of the LST statistical modelling for texture

characterisation and to compare their performances with those obtained by employing different state of the art texture analysis methods.

For each iteration, the reference data is divided into 50% training data and 50% validation data. The data samples belonging to one class are equally divided between the training and validation data. For every sample in the validation set, the distances between its textural signature and the textural signatures of all the training data samples are computed. The distance measure is particular to the texture analysis method employed, as described in Sections 1.4.5, 1.3.5 and 3.4. The validation data sample is assigned a class, according to the decision rules of the supervised classification algorithm employed. The classification performances are evaluated for each iteration and the procedure is repeated 100 times. Finally, the mean classification performances are computed for all iterations [Regniers 2015a].

3.4 State of the Art Texture Analysis Methods

The content based image retrieval and supervised classification performances of the LST based methods are compared against those obtained by different state of the art texture analysis methods. This section starts with a reminder of the different LST statistical approaches proposed here for validation, followed by a brief presentation of the state of the art texture analysis methods considered for comparison in this context.

When it comes to the LST statistical models, their parameters account for the textural signature of the analysed textured image or patch. They divide into two main groups. The first one corresponds to LST statistical models defined on the affine-invariant metric space (see Section 1.3). They consist of:

→ $\mathbf{M}$ - the center of mass, computed in terms of the geodesic distance;

→ $\mathrm{G_{AI}}$ - the Riemannian Gaussian distribution of center of mass $\mathbf{M}$ and dispersion σ;

→ $K\mathrm{G_{AI}}$ - the mixture model of K Riemannian Gaussian distributions of parameters $\mathbf{M}_k$, σ_k and ω_k, $k = 1, ...K$.

The second group consists of LST statistical models defined on the log-Euclidean metric space (see Section 1.4):

→ $\boldsymbol{\mu}$ - the center of mass computed as the Euclidean mean on the LE metric space;

→ $\mathrm{G_{LE}}$ - the 3 dimensional multivariate Gaussian distribution of mean $\boldsymbol{\mu}$ and covariance matrix Σ;

→ $K\mathrm{G_{LE}}$ - the mixture model of K multivariate Gaussian distributions of size 3, of parameters μ_k, Σ_k and ω_k, $k = 1, ...K$.

All the above statistical models are considered for describing marginal LST distributions. However, on the LE space, another model is defined for describing the spatial dependencies inside a local structure tensor field. It consists of - G_{LE_p} - the multivariate Gaussian distribution and the mixture model of K multivariate Gaussian distributions respectively. Its size depends on the size p of the considered pixel neighbourhood configuration, and is given by: $3 \times p$ (see Section 2.3.2 of Chapter 2).

All LST statistical modelling based approaches for texture characterisation are rotation invariant. This property is ensured by applying a transformation to the LST field prior to model parameters estimation, as described in Section 2.4 of Chapter 2.

Secondly, the state of the art texture analysis methods considered for comparison in the different experimental settings presented in this chapter are enlisted below. They can be divided into three groups. The first consist of statistical and so called "traditional" texture analysis methods, widely employed in the literature, with competitive results in spite of a relative low complexity methodology compared to more recent texture analysis approaches. Due to these qualities, they are considered many times as references when assessing the potential of newly proposed texture analysis methods [Regniers 2013, Regniers 2015a, Regniers 2016, Ilea 2015, Pham 2016a, Pham 2016b, Aptoula 2014]. The concerned methods are gray level co-occurrence matrices [Haralick 1973] and local binary patterns [Ojala 1994]:

→ $GLCM_1$ - denotes a rotation invariant version of benchmark GLCM approach [Haralick 1973] for texture analysis. Rotation invariance (RI) is provided by computing the co-occurrence matrices corresponding to 4 different orientations (0°, 45°, 90°, 135°), as previously proposed in [Haralick 1979]. Descriptors are then extracted for all 4 matrices and further averaged [Maillard 2003]. The mean values of each descriptor are concatenated in a feature vector representing the textural signature of the analysed textured image or patch. However, it should be noted than only a *pseudo* rotation invariance is achived in this way. This approach has been previously employed by Regniers et al. in [Regniers 2014a, Regniers 2015a];

→ $GLCM_2$ represents the gray level co-occurrence matrix approach implemented after applying a rotation to the anisotropic textured images or patches so that they all have the same orientation. The rotation is applied after estimating the main orientation of a textured patch by structure tensor computation. The estimation of the rotation angle is given by the first eigenvector of the covariance matrix of gradients [Regniers 2014a]. $GLCM_2$ is, thus, a fully rotation invariant approach;

→ LBP_{ri_1} - correponds to a rotation invariant implementation of the local binary pattern method. Originally proposed by by Ojala et al. in [Ojala 2002b] and denoted there as LBP_8^{ri36} it considers 8 pixels circular neighbourhoods. Each binary pattern obtained for a given pixel neighbourhood is applied a circular bit-wise shift until the minimum of the sequence is attai-

ned. Consequently, the 256 set of possible codes that correspond to the 8 pixel neighbourhood can be reduced to a set of 36 rotation invariant codes;

→ LBP_{ri_2} - stands for the multiscale, uniform and rotation invariant implementation of the local binary pattern approach, originally proposed in [Ojala 2002b] and tested by Aptoula in [Aptoula 2014], where it is denoted by $LBP^{riu2}_{8,1+16,2+24,3}$.

The second set of state of the art texture analysis methods used as a reference in this work is based on the probabilistic modelling of a texture's wavelet subband coefficients. These approaches have been proposed in [Bombrun 2011b, Lasmar 2014], adapted and tested by Regniers et al. [Regniers 2014a, Regniers 2014c] in the same experimental settings as some of those proposed for validation here. Among all the statistical models that have been proposed, we selected for comparison those yielding the best performances:

→ SCM - represents the multivariate Gaussian model with *Sample Covariance Matrix* estimator for the covariance matrix

→ $SIRV_g$ - denotes the SIRV (*Spherically Invariant Random Vectors*) model with a multivariate Gaussian distribution

→ GCG - stands for the multivariate Gamma distribution on Gaussian copula.

For the methods based on the statistical modelling of a texture's wavelet subband coefficients, the rotation invariance is ensured by rotating all the textured image data in a database in order to impose the same orientation to all textures, prior to model parameter estimation [Regniers 2014a].

Last but not least, the LST statistical modelling based approaches for texture characterisation are compared against two texture analysis methods based on spectral decomposition:

→ Gabor - represents a method where the textural features are extracted for each image by applying banks of Gabor filters tuned to different scales and orientations;

→ Aptoula - denotes a rotation invariant multiscale texture analysis approach proposed by Aptoula in [Aptoula 2014]. The periodicity of a texture is firstly described by two feature vectors relying on mathematical morphology methods, namely the circular covariance histogram and rotation invariant point triplets. In addition, the texture is characterized by its degrees of coarseness and directionality. These texture characteristics are each quantified by a feature vector resulted from applying different operations to the Fourier Power Spectrum (FPS) of an image. The approach is made multiscale by applying the FPS analysis methods on a quasi flat zone based multiscale image representation. The resulting 4 feature vectos are further concatenated in order to obtain the textural signature of the image under analysis [Aptoula 2014].

The textural signatures are further used by the different classifiers for evaluating the similarity between different textures. The dissimilarity measure between textural signatures is specific to each texture analysis method. Details for the LST statistical models defined on both AI and LE space are given in Sections 1.3.5 and 1.4.5 of Chapter 1. The dissimilarity measure specific to each state of the art method is given in the *Experimental approaches* paragraph of each experimental setting considered in Section 3.5.

3.5 Results

In this section the classification performances of the different LST statistical modelling based approaches for texture characterisation are given and analysed, for all of the 4 different textured image datasets presented in Section 3.2. LE and AI statistical models for describing marginal LST distributions are tested and compared in terms of performances and computation time as well. LE statistical models applied on joint LST distributions for characterising the spatial dependencies in the LST field are equally tested. In all of the experimental settings proposed here for validating the potential of the the proposed statistical models for LST field characterisation, the classification results of the proposed methods are compared against those corresponding to different state of the art texture analysis methods.

3.5.1 Comparison of AI and LE LST Statistical Models

This section provides a comparison of AI and LE statistical models for marginal LST distributions characterisation. The approaches are tested in a CBIR context on the oyster dataset presented in Section 3.2.1. The retrieval performances of the proposed methods are compared in terms of both accuracy and computational expense.

3.5.1.1 Experimental Approaches

The experimental approaches tested here consist of all the proposed statistical models for characterising marginal LST distributions, on the two metric spaces considered in this work for tensor field description, i.e. AI and LE. When mixture models are employed, a number of 3 mixture components has been considered. This choice is motivated by the study of the theoretical models' fit to observed LST distributions performed in Section 2.3.1 of Chapter 2. The analysis conducted on texture patches belonging to the oyster park dataset showed that, generally, a mixture model of 3 components fits well the observed LST distributions, in the case of both AI and LE models.

The textural signature of a patch is given by the parameters of the employed statistical model. The dissimilarity measure used for computing the distance between the textural signatures of two different patches is specific to the methods employed, as described in Sections 1.3.5 and 1.4.5 of Chapter 1.

The following parameter values have been used for estimating the LST field on the textured patches of this database: $\sigma_G = 0.6, \sigma_T = 1.6$. They have been experimentally tuned. More precisely, systematic retrieval experiments have been performed for different pair values of these parameters and the couple of values giving the best retrieval results has been retained. The tested values belong to a range considered as *optimal*, according to the remarks on the structure tensor parameters choice from Section 2.2.3 of Chapter 2. These intervals considered in this case are: ($[0.4, 2]$ for σ_G and $[1, 6]$ for σ_T).

In the following, the content based image retrieval performances obtained on the VHR remote sensing oyster patch database are given for all methods, in terms of both global and per-class average retrieval rate (see Table 3.1).

3.5.1.2 Performances

The CBIR performances of the different LST field statistical models for texture analysis proposed here are assessed in the following paragraphs in terms of retrieval accuracy and computational time as well.

Retrieval accuracy

We will firstly focus the interpretation of the results in Table 3.1 on the global retrieval performances. First of all, all log-Euclidean based approaches outperform their affine-invariant equivalents, in terms of global retrieval accuracy. Secondly, the superiority of the Gaussian model and Gaussian mixture model to the center of mass approach is shown for both metric spaces. The gap between these models is even more prominent on the LE metric space, quantifying as an 8% overall gain of the Gaussian models (G_{LE} and $3G_{LE}$) over the center of mass approach $\boldsymbol{\mu}$.

When a 3 component Gaussian mixture model is employed, we can see that on the AI metric space it leads to higher performances than when a single Gaussian distribution is considered. However, on the LE metric space the Gaussian and Gaussian mixture model yield similar retrieval performances. Thus, while the use of a Gaussian mixture model is justified on the AI metric space, on the LE space, a single Gaussian distribution seems to be well adapted and sufficient for characterising a texture's LST field. This contrast between the Gaussian models defined on the two metric spaces can be explained by the difference in their numbers of degrees of freedom. Furthermore, these experiments validated the study performed on simulated covariance matrix sample sets in Chapter 1, where for a same given order of the statistical models, higher characterisation capacities haven been notices for the LE model. The first observations noted on simulated data have been confirmed later on in Chapter 2, when the adjustment of the theoretical models to empirical LST distributions has been analysed. The experiments showed that, generally, one Gaussian distribution fits well the empirical LST distributions belonging to relatively simple textures. In contrast, when the AI models have been employed, mixture models have proven to be necessary for a good fit of the empirical distributions.

Table 3.1: Oyster parks patches database retrieval results - the mean and standard deviation of the ARR computed over 100 repetitions of the CBIR protocol on the oyster patches database. Values are given for all classes (AC) and for each oyster texture class: C_1, C_2, C_3. Comparison of AI and LE LST statistical models. The best results of each category are marked in dark red.

Method	AC	C_1	C_2	C_3
M	70.9± 3.7	70.1± 5.9	66.8± 4.4	75.7± 4.2
G_{AI}	73.3± 3.2	69.2± 5.4	72.8± 3.7	77.8± 4.8
$3G_{AI}$	77.1± 2.4	71.0± 4.7	78.8± 2.8	81.5± 4.6
μ	71.3± 3.2	71.1± 4.9	67.3± 4.0	75.6± 4.6
G_{LE}	79.3± 2.9	77.2± 5.8	77.9± 3.3	82.8± 4.6
$3G_{LE}$	79.8± 3.1	74.4± 5.5	80.9± 3.1	84.1± 4.7

Table 3.2: CBIR of oyster textured patches - mean confusion matrix corresponding to the LST model $3G_{LE}$

		Predicted class		
		C_1	C_2	C_3
Real class	C_1	74.4	22.2	3.4
	C_2	9.4	80.9	9.7
	C_3	3.5	12.4	84.1

Several remarks will be made in the following regarding the per-class retrieval-performances. We remind here that C_1 stands for the class of cultivated oyster racks, C_2 corresponds to abandoned oyster fields and C_3 denotes the foreshore class.

First of all, it can be noticed that for all the LST statistical modelling based approaches, the best retrieval performances are obtained for class 3. This is the most distinct class in terms of anisotropy. More precisely it is characterised by a rather isotropic textural pattern as opposed to the first two classes endowed with a more pronounced anisotropy. This is an experimental proof of the LST potential in discriminating texture according to its different levels of anisotropy.

The confusion matrix can provide further information for each class concerning the repartition of the misclassified data between the different classes. To this purpose, the confusion matrix corresponding to the LST LE multivariate Gaussian mixture model is given in Table 3.2.

For the first class of oyster racks, a vast majority of the misclassified textured patches ($\sim$ 22% out of $\sim$ 26%) are assigned to the second class of abandoned oyster fields. Regarding the second class of oyster patches, the misclassified textured patches are equally split between C_1 and C_3. Last but not least, most of the misclassified textured patches belonging to C_3 are found by the algorithm as being more similar to the textural characteristics of patches in class C_2.

These classification errors can be explained in terms of the anisotropy of the textured patches, as most of the confusions are a result of similar anisotropy characteristics of textured patches belonging to different classes. For instance, C_1 and C_3 are the most distant in terms of anisotropy characteristics, with C_1 exhibiting the highest and C_3 the lowest degree of anisotropy. As a consequence, only

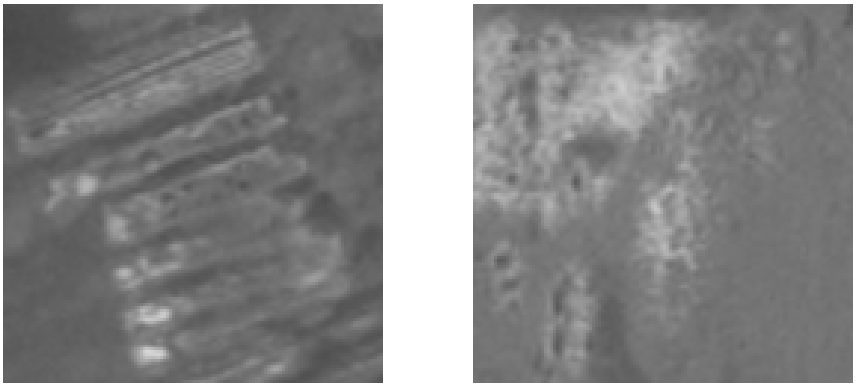


Figure 3.5: Examples of texture patches belonging to class C_2 of abandoned oyster fields: recently abandoned racks (left) vs. unstructured abandoned racks (right)

a small number of misclassified patches of one of these two classes is assigned to the other class. In addition, at a closer look at the database (see Figure A.1 of appendix A), it can be observed that the second class of abandoned oyster fields is the most heterogeneous in terms of texture anisotropy. This class could be subdivided in 2 groups. One group of structured textured patches representing the recently abandoned oyster racks, where the arrangement of the oyster racks is still well marked on the images [Regniers 2014c]. The second group exhibits less anisotropic and more unstructured textural patterns, corresponding to oyster racks that have been abandoned for a longer period of time so that the original alignments of the racks are no longer visible on the image (see Figure 3.5). Thus, the texture patches belonging to the first group of structured abandoned fields is quite similar in terms of anisotropy to the texture patches of class C_1. On the other hand, the patches belonging to the second group exhibit a lower degree of anisotropy, resembling, thus, to the textural pattern of the foreshore class.

Computation time

In this section the different LST statistical modelling methods for texture analysis will be analysed and compared in terms of computational speed.

The computation times relative to the different AI and LE statistical modelling based approaches for LST characterisation are given in Table 3.3. They correspond to MATLAB simulations, performed on a computer of processor Core i7 (3.50 GHz). The given run times include the execution of all the different steps of the CBIR protocol employed on the oyster parks patches database, as previously described in Section 3.3.1. We briefly remind here its main steps. It consists of a first part of signature extraction for all patches in the database followed by 100 iterations of the following steps: new database creation of 25 patches per class - randomly selected from the original dataset, dissimilarity measurements between their textural signatures and evaluation of the mean retrieval performances for the 100 iterations.

There is an important computation time difference between the LST statistical modelling based approaches belonging to the two metric spaces. Log-Euclidean based methods are significantly

Table 3.3: Computation times of LST statistical modelling based approaches: AI vs. LE metric

Method	Center of mass (M vs. μ)	G	3G
Time - LE [min]	0.22	0.35	6
Time - AI [min]	0.43	153	273
Time ratio (AI/LE)	2	437	45.5

faster than those relying on the affine-invariant metric space. This discrepancy is partly due to differences in the complexity of the mathematical operations performed on the two metric spaces. While the LE metric space allows in most cases direct computations, on the AI metric space, most of the mathematical operations are based on recursive estimations.

The simulation time differences between the LE and AI statistical models are specific, naturally, to the LST method employed. The highest computation time ratio between AI and LE based approaches is obtained in the case of the single Gaussian model. This aspect can partly be justified by a difference in the way of measuring the dissimilarity between the textural signatures generated by the two statistical models. In both cases, the symmetric form of the Kullback Leibler divergence - Jeffrey divergence - is used as dissimilarity measure. However, on the AI metric space, there is no analytic form for the Jeffrey divergence and it has to be estimated by means of Monte-Carlo sampling techniques. On the contrary, there is a closed form of the Jeffrey divergence between two multivariate Gaussian distributions on the LE metric space, enabling its direct computation. This difference generates a large gap between the computation times associated to these two statistical models, as the estimation of Jeffrey-divergence implies quite high computational costs.

As far as the Gaussian mixture models are concerned, the time ratio between the AI and LE metric spaces decreases as in this case the Jeffrey divergence has to be estimated for both methods. However, the LE Gaussian mixture model is still significantly faster than its AI homologue, in spite of the fact that a larger sample set is generated for estimating the Jeffrey divergence on the LE metric space (5000 - LE vs. 1000 - AI).

Some remarks can be made regarding the computation times on the LE metric space. Firstly, when modelling the LST field by a single Gaussian distribution, the retrieval protocol is 17 times faster than when using a mixture model of 3 multivariate Gaussian distributions. Despite this run time gap, there is no significant difference in the retrieval performances generated by the two models. This motivates the choice of the multivariate Gaussian distribution as a preferred statistical model for characterising the LST field of a textured image.

The AI and LE LST statistical models have been tested as well on a dataset of maritime pine forest patches extracted from the PLEIADES image described in Section 3.2.1. Those experiments showed an overall 2.5% gain of the LE Gaussian model (see results published in [Rosu 2016]) compared to its AI homologue (see results published in [Rosu 2015a]).

The already long run time required here by the AI Gaussian and Gaussian mixture models would become almost prohibitive in the next context of forest stand supervised classification. This is be-

cause the size of a forest stand texture is much larger than the size of the oyster parks patches tested here. Consequently, the time required for estimating the LST statistical model's parameters for characterising a single forest stand would considerably increase.

3.5.2 Comparison of LE LST Statistical Models and State of the Art Texture Analysis Methods

In this section, LST based methods are compared with a number of state of the art methods in the context of VHR remote sensed image classification. Only the models based on the LE metric are considered here. The databases under consideration are the oyster parks patches and the maritime pine forest stands.

3.5.2.1 Experimental Approaches

Out of the range of statistical models proposed in this work, we choose to uniquely focus here on the capabilities of LE LST statistical models. This choice is justified by the overall gain observed over the AI statistical models in terms of both classification accuracy and computation time, see Section 3.5.1.

The following parameter values were consider on this dataset for structure tensor computation: $\sigma_G = 1.9$ and $\sigma_T = 3$. The parameters have been experimentally chosen by performing repeated experiments around the values considered to be adapted to the textural pattern of the maritime pine forest images. They correspond to a maximum in the classification performances obtained on this database [Kaufmann 2015, Noutatiem Guiafaing 2016].

Some remarks should equally be made on the classifiers parameters choice. Firstly, for the supervised classification algorithm k-NN, all the results presented in Table 3.5 correspond to $k = 5$. The choice of this value is connected to the class being the least represented, namely the class C_4 of clear-cuts. Given that when applying the cross-validation approach, there are only 7 samples in the validation set, the number of nearest neighbours considered cannot exceed this value. Secondly, for the SVM classifier, the standard deviation σ of the Gaussian kernel needs to be adjusted. For every model proposed here a different value was chosen, corresponding to a maximum in the cross-validation classification performances.

A few words will be addressed in the following regarding the parametrisation and specificities of the state of the art texture analysis methods tested for comparison in this experimental context.

For the GLCM_1 approach, four textual descriptors have been extracted from the co-occurrence matrix computed for an interpixel distance equal to 1 - for the oyster parks database and 2 for the maritime pine forest dataset, respectively. The descriptors consist of: correlation, entropy, homogeneity and mean. The inter-pixel distance as well as the combination of these descriptors have been empirically determined by Regniers [Regniers 2014a] and correspond to a maximum in the retrieval performances obtained on each of the two databases. Thus, the textural signature of a patch is com-

posed of a feature vector of size 4. The dissimilarity between textured patches is quantified by the distance between their corresponding textural signatures. The Mahalanobis distance is employed as dissimilarity measure [Regniers 2014a].

We remind here that the $GLCM_2$ differs from $GLCM_1$. More precisely, it consists of an approach that is fully RI. Namely, the co-occurrence matrix has been computed after rotating the anisotropic textured images in the database so that their main orientation is 0. However, it should be stated that the quality of the RI character of the database relies on an accurate estimation of the main orientation of a patch, obtained by structure tensor computation. The orientation of the patch is given by the first eigenvector of the covariance matrix of gradients [Regniers 2014a]. Only anisotropic patches have been applied this rotation, given that isotropic ones have no global dominant orientation. As far as the descriptors are concerned, the same ones as in the case of $GLCM_1$ approach were considered. They correspond to a single orientation between pixel pairs, namely 0°, as opposed to their mean values over 4 orientations, the case of $GLCM_1$.

The textural signature generated by the LBP_{ri_1} approach consists of a feature vector of size 36, corresponding to the discrete occurrence probability density of the rotation invariant binary patterns. The symmetric version of the Kullback-Leibler divergence is used for assessing the dissimilarity between the textural signatures of two different patches.

The textural signature of the two Gaussian models, SCM and $SIRV_g$, used for describing a texture's wavelet subband coefficients is uniquely defined by their covariance matrix. The geodesic distance is used as dissimilarity measure between their textural signatures, given in closed form [Regniers 2014a]. When it comes to the multivariate Gamma distribution on Gaussian copula (GCG), the symmetric version of the analytical expression for the Kullack Leibler divergence between Gaussian copula based multivariate models recently proposed in [Lasmar 2014] is used for assessing the dissimilarity between the textured patches [Regniers 2014a].

3.5.2.2 Performances - oyster parks patches database

In this section the LE LST statistical models are compared against a set of state of the art texture analysis methods in terms of their content based image retrieval performances obtained on the oyster parks textured patches database (see results in Table 3.4). More precisely, the methods chosen for comparison consist of the statistical approaches $GLCM_1$ and LBP_{ri_1} and the following probability models for describing a texture's wavelet sub band coefficients [Regniers 2014a]: GCG, SCM and $SIRV_g$.

When it comes to the comparison of the global retrieval performances between the LST statistical models and state of the art method for texture characterisation, the following remarks can be made. It can be firstly noticed that all the proposed approaches yield higher performances than all the other texture analysis methods considered for comparison, except from GCG and LBP_{ri_1} that are outperformed only by the single Gaussian and the Gaussian mixture models. The gain in retrieval

Table 3.4: Oyster parks patches database retrieval results - the mean and standard deviation of the ARR computed over 100 repetitions of the CBIR protocol on the oyster patches database. Values are given for all classes (AC) and for each oyster texture class: C_1, C_2, C_3. Comparison between methods: upper part of the table - LE LST statistical modelling based approaches, middle part - state of the art statistical texture analysis techniques, lower part - methods based on the probabilistic modelling of a texture's wavelet subband coefficients. The best results of each category are marked in dark red.

Method	AC	C_1	C_2	C_3
μ	71.3± 3.2	71.1± 4.9	67.3± 4.0	75.6± 4.6
G_{LE}	79.3± 2.9	77.2± 5.8	77.9± 3.3	82.8± 4.6
$3G_{LE}$	79.8± 3.1	74.4± 5.5	80.9± 3.1	84.1± 4.7
$GLCM_1$	68.0± 3.1	52.4± 4.9	74.1± 3.3	77.5± 4.5
LBP_{ri_1}	78.2± 2.7	75.6± 4.8	79.7± 3.4	79.2± 4.1
SCM	69.6± 2.9	65.7± 4.5	65.4± 3.7	77.7± 4.9
$SIRV_g$	69.4± 2.6	74.8± 5.2	62.4± 3.4	71.0± 3.7
GCG	73.0± 3.1	74.9± 4.5	67.9± 4.0	76.3± 4.8

accuracies of the LE Gaussian and Gaussian mixture models are around 1% over LBP_{ri_1} and around 7% over GCG.

As far as per class results are concerned, it can be firstly noticed that for all the LST statistical modelling based approaches, the best retrieval performances are obtained for class 3. In the previous section, it has been observed as well that all LST statistical modelling based techniques generated the best retrieval performances for the third class of foreshore. This is the most distinct class in terms of anisotropy. However, for some of the reference texture analysis methods considered here, C_3 is no longer privileged in terms of retrieval score as these methods characterise a texture in terms of other properties and are less efficient than LST based methods in describing the anisotropy information of a texture.

The best retrieval result per-class belongs to LST LE statistical modelling methods, namely G_{LE} for C_1 and $3G_{LE}$ for C_2 and C_3. The second best results per class are obtained by the LBP_{ri_1} approach. For the first class of cultivated oyster racks the performances of the G_{LE} model for LST characterisation slightly exceeds those of LBP_{ri_1}, GCG and $SIRV_g$ (with $\sim$ 2%) while exhibiting a gain higher than 20% over the $GLCM_1$ approach. Regarding the class of abandoned oyster fields (C_2), the performance gain of the $3G_{LE}$ method over LBP_{ri_1} is around 1% and substantially higher (from 6 to 18%) over the other reference methods. As for the foreshore class, the retrieval rate of the LST LE Gaussian mixture model overpasses with 5 to 13% the retrieval scores of the state of the art texture analysis methods.

3.5.2.3 Performances - maritime pine forest stands database

The supervised classification performances relative to grouping the maritime pine forest stands in the different age classes are given in Table 3.5 for all texture analysis methods tested in this experimental

setting.

The proposed methods are compared against LBP_{ri_1}, as this method outperformed all the other state of the art texture analysis methods in the previous experimental context of oyster parks pathes retrieval. In addition, the proposed methods are compared against 3 other state of the art approaches, previously tested in the same experiemental setting (data and classificaion protocol) by Regniers et al. in [Regniers 2014a, Regniers 2014b, Regniers 2015a]. They consist of: $GLCM_1$, $GLCM_2$ and the $SIRV_g$ statistical model for characterising a texture's wavelet subband coefficients.

The GLCM methods have been chosen as reference methods here as in all the experimental settings considered in this work for validating the LST statistical modelling capabilities to characterise texture since the coocurrence matrix can be considered as a benchmark in the texture analysis literature. As for the $SIRV_g$ approach, it has been chosen for comparison since it generated the best classification performances out of the group of statistical models of wavelent subband coefficients exploiting uniquely the spatial dependencies of the panchromatic band of a PLEIADES image, regardless of multispectral information (see more details on these methods and the results generated by other statistical models for wavelet coefficients in [Regniers 2014b]).

For each of the proposed statistical models for LST description a distance measure is proposed in order to compare the probabilistic models characterising different LST fields. The existence of a dissimilarity measure between the LST statistical models employed here as signatures for the forest stands under analysis facilitates the implementation of different classifiers. The *k*-NN and SVM supervised classification algorithms have been employed here for classifying the maritime pine forest stands in the different age classes. The results in Table 3.5 are grouped in two parts, one corresponding to each of the two classifiers. The SVM classifier yields better performances than *k*-NN for all the texture analysis methods tested in this experimental context. The gain in performances is specific to each method and it ranges from 1 to 3%. In the following, the classification results generated by the SVM classifier will be analysed in more detail, as the general conclusions that can be drawn for the two classifiers are quite similar.

A few words will be firstly addressed on the global classification performances, assessed in terms of OA and *kappa* index. The LST statistical model performing the best here is the Gaussian mixture model. The ground-truth and classification maps generated by this method in one iteration of the cross-validation algorithm are displayed in Figure 3.6. A 2% gain is observed over the multivariate Gaussian model, as opposed to the previous oyster patches CBIR experiments, where the two methods generated retrieval results of the same order. However, as in the previously considered experimental context, both LE Gaussian and mixture of Gaussian models yield considerably better performances than the LE center of mass. The classification results generated by the $3G_{LE}$ are of the same order as those obtained by the $SIRV_g$ model, and $\sim 1\%$ higher than those obtained by $GLCM_2$, the best and second-best performing approaches among the state of the art methods considered here for comparison. When it comes to the $GLCM_1$ approach, its classification performances are almost 10% lower than those corresponding to the LE Gaussian mixture model. Although the $GLCM_1$ de-

Table 3.5: Maritime pine forest stands supervised classification results k-NN (upper table) vs. SVM (downward table) classifiers: mean performances computed for 100 iterations of the cross-validation algorithm - the global performances given by OA and κ indicators and the per-class resuls assessed by PA and UA criteria. Comparison between methods: upper part of the table - LST statistical modelling based approaches, middle part - statistical texture analysis techniques (GLCM [Regniers 2014a] and LBP_{ri_1}), lower part - probabilistic modelling of wavelet subband coefficients based methods [Regniers 2015a]. The best results of each category are marked in dark red.

- Results obtained by the k-NN classifier:

Method	OA	κ	C_1	C_2	C_3	C_4	
μ	67.4	0.53	86.2	49.3	68.8	46.9	PA
			60.0	51.7	65.3	91.0	UA
G_{LE}	72.1	0.59	83.6	62.6	76.1	45.3	PA
			60.2	62.7	70.3	92.8	UA
$3G_{LE}$	73.8	0.62	83.9	65.0	80.4	43.0	PA
			59.3	68.9	70.7	91.6	UA
$GLCM_1$	64.4	0.57	70.3	61.2	65.9	60.6	PA
			77.6	55.9	60.2	73.2	UA
$GLCM_2$	75.3	0.65	82.9	72.3	75	54.6	PA
			85.6	72.2	68.9	71.0	UA
LBP_{ri_1}	68.9	0.56	85.3	64.4	60.2	30.0	PA
			63.1	65.9	64.3	82.6	UA
$SIRV_g$	75.2	0.65	83.1	69.0	82.1	47.4	PA
			88.8	73.2	73.2	80.1	UA

- Results obtained by the SVM classifier:

Method	OA	κ	C_1	C_2	C_3	C_4	
μ	69.7	0.56	94.0	51.1	65.6	44.6	PA
			59.1	54.5	69.1	86.7	UA
G_{LE}	75.3	0.64	92.6	64.6	80.2	28.3	PA
			54.1	72.2	75.6	84.3	UA
$3G_{LE}$	77.3	0.67	92.9	67.9	84.5	26.5	PA
			54.6	77.7	76.1	85.3	UA
$GLCM_1$	68.1	0.53	74.8	71.9	64.9	34.3	PA
			76.4	60.0	70.0	61.0	UA
$GLCM_2$	76.0	0.65	89.7	73.3	71.5	39.4	PA
			79.3	72.9	77.8	64.2	UA
LBP_{ri_1}	70.1	0.56	94.9	63.7	57.7	18.6	PA
			49.4	63.3	70.8	86.0	UA
$SIRV_g$	77.4	0.67	91.9	79.2	67.2	34.6	PA
			81.9	73.2	76.8	86.6	UA

notes a rotation invariant implementation of the co-occurrence matrix approach, it can be observed that when computing the co-occurrence matrix on the stands of same orientation ($GLCM_2$), a notable classification gain is obtained ($\sim$ 8%). This aspect stands as a proof of the shortcomings of rotation invariant texture analysis methods that, in most cases, can provide only a *pseudo* rotation invariance, as opposed to the LST techniques that have an intrinsic way of addressing this need.

Figure 3.6: Ground-truth (top) and classification maps (bottom) generated in one iteration of the cross-validation algorithm by the LST statistical model $3G_{LE}$; the training stands are emphasised by marking their contours according to the class color code displayed in the legend while the validation ones are completely filled with the color of their class.

Last but not least, it is interesting to notice that the state of the art texture analysis methods show a more unstable behaviour than the methods based on the statistical modelling of a texture's LST field. More precisely, $SIRV_g$ - the method yielding the best classification performances on the maritime pine forest stands database - generated one of the lowest retrieval results on the oyster patches database, around 10% lower than those corresponding to G_{LE} and $3G_{LE}$ LST models. In addition, the state of the art method performing the best in the previous experimental context, namely LBP_{ri_1}, yields a quite weaker classification performance, approximately 7% lower than the $3G_{LE}$ approach. Nevertheless, in both experimental contexts presented in this section, the LE multivariate Gaussian and the mixture of multivariate Gaussian models for characterising LST distributions yield ones of the most competitive classification and retrieval scores.

The per-class classification performances will be analysed in the following. We remind here that

the PA criterion indicates the proportion of ground-truth images of a given class that are assigned by the classification algorithm to the correct class. On the other hand, the UA indicates the percentage of images among those assigned by the classifier to a given class, that actually do belong to that class, according to the ground-truth.

The first age class ($0-9$ years old) is the one with the highest attained classification scores by all methods tested here. The highest score is obtained by the LBP_{ri_1} method, followed by the LST center of mass model. However, both methods yield average UA scores, around $\sim 50\%$ for the LBP method and moderately higher ($\sim 60\%$) for the $\boldsymbol{\mu}$. While most of the C_1 forest stands have been correctly classified, these scores indicate that many other forest stands assigned by the classifier to this class actually belong to other classes thus decreasing the confidence the user of the classification maps can assign to these results. Probably the best classification compromise between the producer's and user's accuracy for this class is provided by the SIRV_g model, where the over 90% classification accuracy can be viewed with a significantly higher degree of confidence given that the UA score is above 80%.

The second-highest classification scores per class for all LST based methods are obtained for C_3 (forest stands of age $>$ 20 years old) as opposed to the state of the art methods that all yield their second-highest scores for the forest class C_2 ($10-19$ years old) (see the results generated by the SVM classifier in Table 3.5). The Gaussian mixture model performs the best on class C_3 with a 13% gain over the second most performant method for this class, namely GLCM_2. The classes C_1 and C_3 are the most contrasting in terms of the anisotropy characteristics of their textural patterns. As in the case of the previous experimental context, these high classification scores associated to the most distant classes in terms of anisotropy, reveal a finer capacity of LST based methods for anisotropy characterisation, as opposed to the state of the art texture analysis methods.

The best PA score for C_2 are generated by the state of the art SIRV_g method at an around 11% accuracy distance from 3G_{LE} - the LST statistical modelling based method yielding the highest classification scores on this class. However, the classification results generated by 3G_{LE} are a little more reliable than those corresponding to the SIRV_g model, provided the $\sim 4\%$ gain in terms of the user's accuracy indicator.

As for the last class of clear-cuts, it presents the lowest classification scores among the 4 maritime pine forest stands classes, for all methods. The highest accuracy ($\sim 45\%$) is generated by the LE center of mass with a user's accuracy of $\sim 87\%$. Among the state of the art methods, GLCM_2 and SIRV_g yield similar performances, with PAs of $\sim 40\%$ and $\sim 35\%$ respectively. However, more reliability can be given to the SIRV_g results, since the UA score is about 87%, more than 20% higher than the UA corresponding to the GLCM_2 approach.

The classification results per class are further investigated in terms of common classification errors and inter-class confusion. To this purpose, the confusion matrix for the LST statistical model 3G_{LE} is displayed in Table 3.6.

By analysing the confusion matrix results, it can be noticed that the majority of misclassified

Table 3.6: Supervised classification of maritime pine forest stands - mean confusion matrix corresponding to the LST model $3G_{LE}$ and generated by the SVM classifier

		Predicted class			
		C_1	**C_2**	**C_3**	**C_4**
Real class	**C_1**	92.9	5.4	0.0	1.7
	C_2	10.3	67.9	21.8	0.0
	C_3	1.1	14.4	84.5	0.0
	C_4	67.4	0.6	5.5	26.5

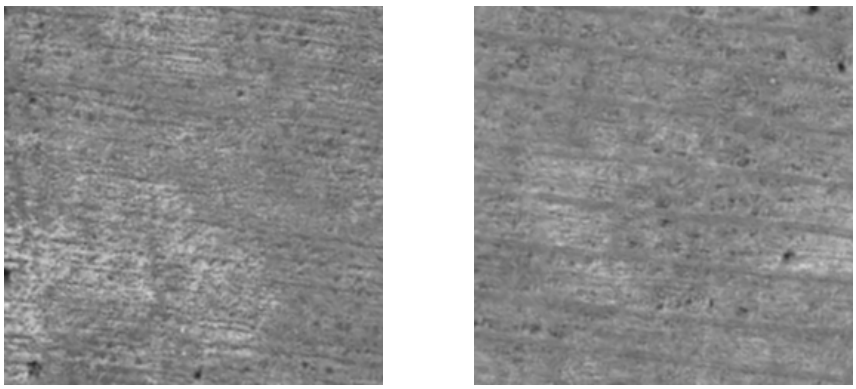


Figure 3.7: Examples of texture patch extracts from maritime pine forest stands of class C_1 (left) and C_4 (right)

forest regions belonging to the first age class were assigned to class 2, a few to C_4 and none to C3. This result is easily explained in terms of the anisotropy level of the textured samples. Classes C_1 and C_2 are the classes with the most pronounced level of anisotropy, enforcing thus a strong confusion between the samples belonging to these two classes. In addition, some of the patches in C_4 present as well anisotropic textural patterns. On the contrary, the most distant class from C_1 in terms of anisotropy is C_3, explaining thus why none of the misclassified texture samples of C_1 were attributed to C_3.

For the second age class, the misclassified regions are divided between the classes C_1 and C_3, this results being coherent with the anisotropy considerations previously mentioned. As far as the third age class (> 20 years old) is concerned, most of the classification errors have been made in the second age class (10 and 19 years old).

The results corresponding to the clear cuts class are quite particular. There is a higher percentage of samples assigned to a different class (namely $\sim 67\%$ of the ground-truth data is classified to C_1) than the actual percentage of correctly classified samples (as little as $\sim 27\%$). These classification errors can be explained, to some extent, by taking a closer look to the image content in these classes. There are some regions on the PLEIADES satellite image with no visible vegetation, but with discernible ploughing machines traces on the field. These traces may be seen as a reforestation sign and, as a consequence, the samples taken from these regions belong to the first age class. However, they are visually very similar to the clear-cuts samples, inducing thus a high predisposition to confusion

(see an example of texture patch extracts in Figure 3.7). In addition, the textural information of the clear cuts class might be difficult to capture in its full complexity, given the quite limited number of ground-truth samples (7).

Another remark should be made on the nature of this database, that can partly justify the between-class confusion. To be specific, the age is a continuous variable and its discretization may be responsible for some of the classification errors, especially for those forest stands of age close to the border age separating two classes [Regniers 2014a]. Moreover, according to [Regniers 2015a], 15% of the reference data is composed of outliers. As a consequence, the global classification accuracy can hardly exceed 85%.

3.5.3 Evaluating Rotation Invariance and the Relevance of Joint LST Statistical Models.

The focus of this section is dual. One objective consists of evaluating LE joint statistical models for characterising the spatial dependencies inside the LST field of a textured image. Moreover, the intrinsic quality of LST based approaches to deal with rotation invariance is experimentally validated, as the test database consists of anisotropic textured patches of randomly imposed orientations. These aspects are investigated in a CBIR context of carbonaceous material patches of snapshots issued from HRTEM technology.

3.5.3.1 Experimental Approaches

For the LST statistical modelling based methods for texture characterisation, the structure tensor has been computed with the following parameters values: $\sigma_G = 1$, $\sigma_T = 3$. They correspond to a maximum in the retrieval performances obtained on this database, after performing systematic tests for values considered adapted to the textural pattern of these images.

The log-Euclidean LST statistical modelling approaches tested in this experimental context consist of: the center of mass $\boldsymbol{\mu}$, a multivariate Gaussian distribution $\mathrm{G_{LE}}$ of size 3 for characterising marginal LST distributions and a multivariate Gaussian distribution of superior size $\mathrm{G_{LE\_p}}$ for characterising distributions of structure tensors p-tuples encompassing the LST field's spatial dependencies information. Several neighbourhood configurations have been tested, namely pixel couples, cross and square neighbourhoods of different sizes. The inter-pixel distance has been varied between 1 and 15. The $G_{\mathrm{LE\_p}}$ model results in Table 3.7 corresponds to the neighbourhood configuration maximising the retrieval results: cross-shaped neighbourhood of size 3 and inter-pixel distance equal to 8 (see the neighbourhood topologies illustrated in Figure 2.17 of Chapter 2).

Mixture models of multivariate Gaussian distributions have not been considered here, since in the previous experimental settings presented in this study (see Tables 3.1 and 3.5) the Gaussian model proved to be well-adapted and sufficient for characterising the LST information of a textured image. This aspect has also been observed in some other experiments conducted by the authors

on different datasets, that make not the object of this report. In addition, the same observations have been made when analysing the fit of the theoretical distributions to empirical LST fields (see Section 2.3.1 of Chapter 2). Moreover, the LE multivariate Gaussian model is 17 times faster than its corresponding mixture model of 3 multivariate Gaussian distributions, as shown in Section 3.5.1.

The retrieval performances obtained by the proposed approaches are compared against those obtained by the RI implementation of the gray level co-occurrence matrix method, $GLCM_1$. A distance of 1 was considered between the pairs of pixels. The textural signature is given by the mean value of the following descriptors: homogeneity, entropy and correlation, corresponding to the 4 different orientations chosen for this method.

3.5.3.2 Performances

Table 3.7: Carbonaceous material patches database retrieval results - The ARR values values are given for all classes (AC) and for each HRTEM carbon material texture class: C_1, C_2, C_3, C_4. The best results are marked in dark red.

Method	AC	C_1	C_2	C_3	C_4
μ	70.8	60.9	90.6	89.1	42.6
G_{LE}	86.0	80.1	100	98.4	65.6
G_{LE_p}	88.4	86.3	100	98.4	68.8
$GLCM_1$	68.2	80.5	65.2	78.5	48.4

As seen in the previous experiments, the retrieval results in Table 3.7 confirm one more time that the LE multivariate Gaussian model is far more adapted for LST characterisation than the simple use of the LST field's center of mass. When including the LST spatial dependencies information, by the use of an extended multivariate Gaussian distribution that characterises neighbourhoods of LE structure tensors, there's a slight improvement in the global retrieval performances (2% gain over the G_{LE} model). The general trait observed during different experiments on various textured data, is that the model for describing LST spatial dependencies generally provides same order results as the G_{LE} model for characterising marginal LST fields. The improvements are really insignificant, as observed here and in the context of CBIR of maritime pine forest patches, where a 1% gain has been observed (see results in [Rosu 2016, Noutatiem Guiafaing 2016]). The LE statistical models have been applied for modelling multiscale dependencies in LST fields as well, with no improvements in terms of texture recognition performances [Noutatiem Guiafaing 2016].

In terms of retrieval results per class, the best performances of LST statistical modelling based methods are obtained on this database for C_2 and C_3, in spite of a quite strong visual similarity of the textures belonging to these classes. The texture patterns of carbon material patches in classes C_1 and C_4 might be more accuratly described when employing a larger-scale analysis, as suggested in [Da Costa 2015].

As far as the rotation invariance is concerned, the LST statistical modelling based techniques yield relatively high retrieval scores, in spite of the random orientation of the anisotropic textured

patches in this database. Thus, the intrinsic ability of the LST based techniques to discriminate textures according to their pattern and anisotropy information no matter their orientations is experimentally validated by these results. When it comes to the co-occurence matrix global result, it shows a performance loss of $\sim$ 20% compared to the G_{LE_n} model for LST field description. This result experimentally highlights the quite limiting abilities of $GLCM_1$ approach of addressing the rotation invariance need of a classification application of anisotropic textured images.

3.5.4 Benchmarking LST Based Methods in Remote Sensing Literature

The following section deals with the last experimental setting considered for evaluating the potential of the proposed statistical models for LST field characterisation and considered for presentation in this report. It consists of a CBIR application of both textured and non-textured remote sensing patches of the publicly available database LULC of UC Merced [Yang 2013]. LE marginal statistical models are tested here against other state of the art texture analysis methods previously tested on this database. The experiments performed on this collection gives access and thus facilitates the comparison to other image analysis methods used for dealing with remote sensing images. Our main aim here resides in placing the LST statistical modelling based approaches for texture analysis on the much wider map of image analysis methods applied in the context of remote sensing imaging.

3.5.4.1 Experimental Approaches

The LST statistical modelling based approaches tested here are those relying on the log-Euclidean metric space, namely: $\boldsymbol{\mu}$, G_{LE} and $3G_{LE}$. The structure tensor has been computed on this dataset for the following parameter settings: $\sigma_G = 1.6$, $\sigma_T = 5.1$. As for the other databases, they had been hand-tuned by performing systematic tests in a range of values considered to be adapted to the dominant textural patterns of the patch collection, according to the considerations explained in Section 2.2.3.

Among all state of the art image analysis previously tested on this database [Yang 2013, Aptoula 2014, Özkan 2014], we focus here uniquely on the texture analysis methods, since we considered they are the only ones making the object of a fair comparison. They consist of the following approaches: Gabor, LBP_{ri_2}, Aptoula. In the following, some details concerning the implementation of each of these methods in the context of a CBIR application are given.

As far as the texture analysis method relying on Gabor filtering is concerned, banks of filters tuned to 5 scales and 6 different orientations have been used. The mean and standard deviation is computed for each filter, resulting, thus, in a feature vector of size 60 [Yang 2013]. The similarity measure between images is computed by a modified version of the L_2 distance function between their corresponding feature vectors, in order to achieve rotation invariance (for more details, see [Yang 2013]).

LBP_{ri_2} consists of the multiscale, uniform and rotation invariant implementation of local binary

Table 3.8: Retrieval results on LULC patches database. Global performance indicators are given in terms of the ANMRR. Left side - LST statistical modelling based approaches, right side - state of the art texture analysis methods. The best results of each category are marked in dark red.

Method	ANMRR	Method	ANMRR
μ	0.752	**Gabor** [Yang 2013]	0.630
$\mathbf{G_{LE}}$	0.558	$\mathbf{LBP_{ri_2}}$ [Aptoula 2014]	0.735
$\mathbf{3G_{LE}}$	0.565	**Aptoula** [Aptoula 2014]	0.575

pattern approach tested previously on the same database in [Aptoula 2014]. It generates as textural signature a feature vector of size 54, corresponding to the normalized histogram of the 54 different LBP outputs for this approach. Five different dissimilarity measures are tested by Aptoula in [Aptoula 2014], namely: the Euclidean, Manhattan, Intersection, χ-square and Bhattacharyya distance between the textural signatures of 2 patches. The $\mathrm{LBP}_{\mathrm{ri}_2}$ results in Table 3.8 correspond to Bhattacharyya distance, the one yielding the best retrieval results among this group of 5 different dissimilarity measures.

The last texture analysis state of the art method considered for comparison in this experimental setting, denoted by Aptoula, consists of the concatenation of feature vectors resulting from morphological texture descriptors and two other texture descriptors based on the Fourier power spectrum image analysis. Two image scales of analysis have been considered and the textural signature of a patch is given by a feature vector of size 62. The same set of 5 disimilarity measures has been tested as in the case of the $\mathrm{LBP}_{\mathrm{ri}_2}$ approach. The results given for this method in Table 3.8 correspond to the χ-square distance, the one maximising the retrieval performances of this method on the LULC database [Aptoula 2014].

3.5.4.2 Performances

The overall retrieval performances for this database are assessed in terms of the ANMRR and given in Table 3.8. We remind here that the ANMRR retrieval performance indicator takes values between 0 and 1, with a lower score showing higher retrieval performances. It should be noted that this indicator was computed while considering 200 images as relevant to retrieval, twice the number of images per class.

As far as the LST statistical models are concerned, a single multivariate Gaussian distribution outperforms the mixture model of 3 such distributions. Moreover, both $\mathrm{G_{LE}}$ and $\mathrm{3G_{LE}}$ outperform the state of the art texture analysis methods considered here for comparison, in spite of a lower complexity compared to other approaches such as Aptoula. To the authors' knowledge, this selection reunites all the state of the art texture analysis methods that have been tested to this day on this database (see Table 3.8). However, the highest retrieval performances on this database are not generated by texture analysis approaches but by various image representation and simplification techniques of local invariant features. Thus, the best retrieval score obtained so far (0.4505) belongs to the image

simplification tool VLAD-PQ (product-quantized binary version of the vector of locally aggregated descriptors) applied on SIFT (scale-invariant feature transform) descriptors - [Özkan 2014].

3.6 Conclusions

The main objective of this chapter has been the experimental validation of the LST statistical models for LST field characterisation. The proposed methods have been tested in different contexts of classification of various types of textured data. In addition, content based image image retrieval experiments have been conducted on non-textured data as well. Two main types of imagery have been concerned, namely VHR remote sensing and material lattice fringe imagery issued from HRTEM technology.

Several LST statistical modelling based approaches have been tested, defined on the affine-invariant and log-Euclidean metric spaces. The majority of the proposed methods are adapted to the description of marginal distribution of tensors. For the AI metric space, the statistical modelling based methods consist of the center of mass, the AI Gaussian distribution and its corresponding mixture model. Similarly, the statistical modelling based approaches on the LE metric space are as follows: the LE center of mass, the multivariate Gaussian distribution and its corresponding mixture model. In addition, on the LE metric space, the statistical modelling of p-tuples of structure tensors by a multivariate Gaussian distribution, encompassing as well the LST spatial dependencies information, has equally been proposed.

The performances of the methods proposed in this work have been compared to those generated by different state of the art texture analysis methods. They consist of statistical methods, namely different implementations of the co-occurrence matrix and local binary patterns approaches, methods based on the probabilistic modelling of a texture's wavelet subband coefficients and on texture analysis methods based on the spectral decomposition. All proposed methods as well as the reference ones are rotation invariant.

Among the statistical modelling based approaches proposed here for LST characterisation, the ones relying on the LE metric space have proved to be the most interesting. Not only that the LE statistical models are significantly faster than their AI equivalents, but in general they either outperform or yield similar performances as the AI statistical models. In addition, a multivariate Gaussian model on the LE space generated better or same order results as its corresponding 3 component mixture model, while being significantly faster. Thus, the use of a mixture model for LST field characterisation does not present much interest on the LE metric space. In contrast, on the AI metric space, the use of a mixture model generally improves the performances. However the gain in classification accuracy comes at a quite high computational expense. Moreover, when modelling neighbourhoods of structure tensors to describe the spatial dependencies inside the LST field, moderate performance improvements have been observed with respect to the case when marginal LST distributions are modelled.

LST statistical modelling techniques have proved to outperform, in most cases, the state of the art texture analysis methods proposed for comparison. In addition, they show a much more stable behaviour, since they produce competitive results in all the experimental settings presented in this chapter, in spite of the difficulties arising in the different databases. In contrast, the performances of the state of the art methods vary from one application to the other. Thus, if some methods yield competitive results in a given experimental context, they showed an important decrease in performances in other test scenarios. These methods seem to depend more on the nature of the textured data under analysis, showing thus less versatility than the LST statistical modelling based approaches.

Finally, the intrinsic ability of LST based approaches to respond to the rotation invariance need of a classification application of anisotropic textures, has been experimentally proven as well. The LST statistical modelling based approaches provide high performances even on datasets of patches of random orientations, as opposed to the rotation invariant implementations of the state of the art texture analysis methods. The later proved their shortcomings in addressing this need, providing, thus, a rather *pseudo* rotation invariance.

Chapter 4

Synthesising LST Fields with LE Models

"When in doubt, smooth."

–Sir Harold Jeffreys, Mathematical Statistician

Contents

4.1 Introduction and Objectives

The statistical models proposed in Chapter 1 have showed promising LST field characterisation potentials, in the texture recognition experiments presented in Chapter 3. The main objective of this chapter is to further evaluate the statistical models' potentials in the case of higher complexity tasks that require richer characterisation capacities than the task of texture recognition. More precisely, the potentials of the statistical models to synthesise LST fields is assessed in the following.

The principle of the LST field synthesis approach proposed in this work is illustrated in Figure 4.1. An input LST field is given as starting point. It can be represented by the observed LST field of a real texture or by a randomly generated LST field, as well. The LST synthesis consists in reproducing an output LST field very close in characteristics to the input LST field. The similarity between the input and output LST fields can be visually assessed by comparing the input and output maps of the LST parameters. While this is not the main objective of the present work, LST field synthesis could further be used to the purpose of texture synthesis, as it has previously been done by Akl et al. [Akl 2018]. The goal of texture synthesis is to create an output texture having the same visual characteristics as the input texture.

The LST synthesis framework proposed in this work is developed on the LE space and is based on parametric methods. More precisely, the input LST field is described by a LE statistical model (multivariate Gaussian model or its corresponding mixture model). A Markovian hypothesis is made and the synthesis of the output LST field is uniquely based on the estimated parameters of the statistical model chosen for describing the input LST field.

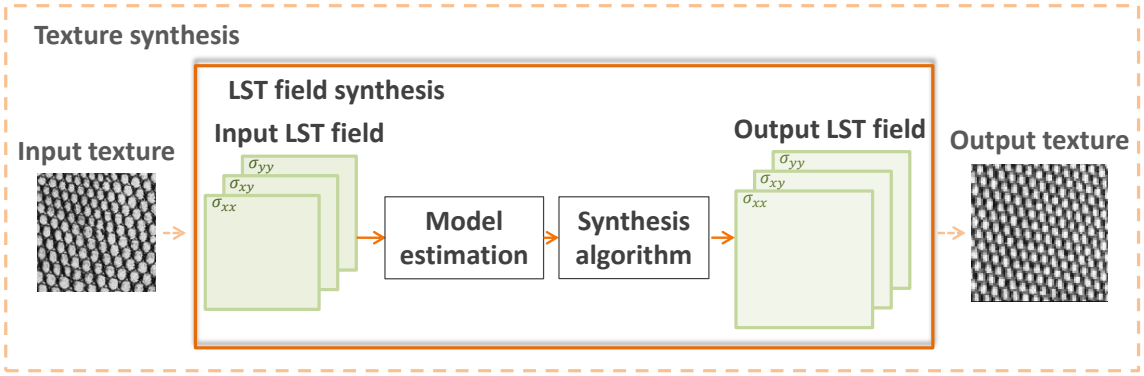


Figure 4.1: LST field statistical modelling based synthesis scheme - stepping stone to texture synthesis.

4.2 Related work

The topic of LST field synthesis has rarely been addressed in the litterature [Akl 2016, Akl 2018]. In these studies, the LST field synthesis is used in a two steps texture synthesis process. More precisely, the texture's structure layer is first built, by synthesising an LST field similar to the LST field

computed on the example texture. The synthesised field is further used to constrain the second step of texture synthesis. A multiscale non-parametric method is proposed for the LST field synthesis, inspired by the texture synthesis method proposed by Wei et Levoy [Wei 2000].

The state of the art parametric LST field synthesis approach proposed by Akl et al. [Akl 2016, Akl 2018] relies on the definition of a neighbourhood of structure tensors, in each point of the image. The structure tensors neighbourhood is composed of the set of 2×2 structure tensors corresponding to the positions in the considered neighbourhood topology. Similarly, for each point in the output LST field, a neighbourhood of structure tensors is defined. The output LST neighbourhood is compared to all the input LST neighbourhoods (the LST neighbourhoods corresponding to all the points in the input texture). Different metrics, adapted to the particularities of non-negative definite matrices are considered for measuring the similarity between individual structure tensors. The similarity between two structure tensor neighbourhoods is evaluated by computing the sum of structure tensors dissimilarities over the entire neighbourhood. The current position in the output LST field is assigned the structure tensor in the input LST field minimising the sum of structure tensor similarities. The pixel by pixel synthesis is repeated several times in the aim of obtaining an output LST field close to the input one.

4.3 LST Synthesis - Principle and Algorithms

We propose in this work a parametric Markov Random Field (MRF) approach for LE LST field synthesis. A Markovian hypothesis is made by considering neighbourhoods of vector-forms of LE mapped structure tensors in each point of the LST field, in order to characterise the spatial dependencies. The extended input LE LST field formed in this way is characterised by a statistical model from the LE family (multivariate Gaussian distribution or multivariate Gaussian mixture model - see Section 2.3.2.2 of Chapter 2). The section starts with some key theoretical considerations on the MRF synthesis and on the relaxation algorithms employed. The adaptation of these concepts to LST field synthesis on the LE metric space is explained at the end of this section.

4.3.1 Markovian Synthesis

Markov Random Fields are largely employed for solving computer vision problems of different natures. Some examples are image restoration, reconstruction and segmentation, texture analysis and synthesis, edge detection. The approach relies on describing the underlying image geometry by modelling the local spatial dependencies. Pioneering research has been conducted since the $70's$, establishing thus the fundamentals of MRFs in image processing [Besag 1974, Geman 1984, Besag 1986].

The MRF approach in image processing is defined in terms of sites and labels. A site is a point in the space whereas a label is an event than might happen to a site [Li 2001]. The MRF approach has been adapted here from images to LE LST fields. Considering the LST field computed in each

point of an image $\mathbf{I}$ of size $[w \times h]$, its corresponding set of sites is given by $\mathbf{Z} = \{z_1, ...z_{w\times h}\}$. The vector-form value associated to each site is given by a 3 component vector label x_i in the state space $\mathbf{\Lambda} = \{\lambda_1, ...\lambda_l\}$, where l corresponds to all possible states of the LE mapped structure tensors. The labelling process corresponds to assigning a label from the state space $\mathbf{\Lambda}$, to each site in $\mathbf{Z}$. In practice, the set of possible states corresponds to the LE LSTs in the input LST field. In some cases that are detailed later in this work, the original state space is further quantified in order to reduce its size.

4.3.1.1 Markovian Hypothesis

The MRF approach relies on the characterisation of the inter-relationship between sites which is defined by considering a neighbourhood system. Different neighbourhood topologies can be defined, including the examples previously illustrated in Chapter 2 (Figure 2.17).

Let $X = \{X_1, ...X_{w\times h}\}$ be a random field of 3 component vectors (corresponding to vector-forms of LE mapped structure tensors) defined on $\mathbf{Z}$. Each random vector X_i is associated a state x_i in $\mathbf{\Lambda}$. The following notations are employed. The event that the random vector X_i receives the label $x_i \in \mathbf{\Lambda}$ is denoted by $X_i = x_i$. In addition, the probability associated to the event is denoted by $P(X_i = x_i)$. Similarly, the notations $X = x$ and $P(X = x)$ are employed for the joint event, where $x = \{x_1, ..., x_{w\times h}\}$ is called a configuration of X. It corresponds to a realization of the random field. If the same label set is associated to each site, the configuration domain (the set of all possible configurations) is given by: $\mathcal{D} = \mathbf{\Lambda}^{w\times h}$.

The following two conditions need to be satisfied so that X is a Markov Random Field on $\mathbf{Z}$:

$$P(x) > 0, \forall x \in \mathcal{D} \quad \text{(positivity)} \tag{4.1}$$

$$P(X_i = x_i | X_j = x_j, j \neq i) = P(X_i = x_i | X_j = x_j, j \in V_i) \quad \text{(Markovianity)} \tag{4.2}$$

with the conditional probability at the site i being called the *Local Conditional Probability Density Function* (LCPDF). The first property states that the probability of a given label configuration is always positive. The Markovianity property states that the random vector X_i describing the structure tensor's state at any site in $\mathbf{Z}$, can take any label x_i from $\mathbf{\Lambda}$, but the probability that $X_i = x_i$ uniquely depends on the labels x_j at sites z_j neighbouring z_i.

The proposed method consists of a site by site LE LST synthesis based on maximising the likelihood of the label x_i at the site z_i, in terms of the local conditional probability density function. To this purpose, relaxation algorithms are employed, so that: $x_i \leftarrow \arg\max_{\lambda\in\mathbf{\Lambda}} P(\lambda, z_i)$.

Algorithm 1 ICM (Iterated Conditional Modes)

Input:

Niter - number of iterations of the relaxation algorithm
$\mathbf{Z} = \{z_1, ...z_{w\times h}\}$ - set of sites
$\mathbf{\Lambda} = \{\lambda_1, ...\lambda_l\}$ - state space
$x_0 = \{x_1, ...x_{w\times h}\}$ - initial label configuration for the sites in $\mathbf{Z}$
P - local conditional probability density function

Output:

$x = \{x_1, ...x_{w\times h}\}$ - output label configuration for the sites in $\mathbf{Z}$

Begin

1: iter $\leftarrow$ 1
2: $x \leftarrow x_0$
3: **while** iter $\leq$ Niter **do**
4: **for** each site z_i in $\mathbf{Z}$ **do**
5: $x_i \leftarrow \arg\max_{\lambda\in\mathbf{\Lambda}} P(\lambda, z_i, x)$
6: **end for**
7: iter $\leftarrow$ iter + 1
8: **end while**

End

4.3.1.2 Relaxation Algorithms

Several relaxation algorithms can be employed for the LST synthesis process. The objective is to obtain label configurations maximising the joint probability of the MRF. This is achieved in an iterative way. At each iteration, a site by site synthesis is achieved by maximising the LCPDF in each site in $\mathbf{Z}$. The relaxation algorithms employed here consist of *Iterated Conditional Modes* (ICM), *Metropolis* and *Metropolis Simulated Annealing* (Metropolis SA).

ICM

ICM consists of an iterative deterministic relaxation technique [Besag 1986]. In each site, an exhaustive search is done in the state space $\mathbf{\Lambda}$ and the label maximising the LCPDF is retained. Generally, the algorithm converges to a local maximum after a small number of iterations [Won 2004]. However, as for each site all labels are tested, one iteration (corresponding to a complete scan of the sites) can take quite long, depending, naturally, on the size of the state space. The ICM synthesis process is given in the Algorithm 1.

Metropolis and Metropolis Simulated Annealing

Metropolis algorithm [Metropolis 1953] is employed as well in this work for LE LST field synthesis. It is an iterative stochastic relaxation algorithm, part of Monte Carlo Markov Chain (MCMC) techniques. At each iteration, for a given site z_i, a label is randomly sampled from the state space $\mathbf{\Lambda}$. The ratio between the probabilities of the new label proposition λ and the current label x_i is computed. A random probability value is further sampled in the interval $[0, 1]$. The new label proposition λ is retained for the current site, if the ratio between its LCPDF and the LCPDF of x_i (current la-

Algorithm 2 Metropolis Simulated Annealing

Input:

Niter - number of iterations of the relaxation algorithm
k_{temp} - temperature reduction coefficient
$\mathbf{Z} = \{z_1, ...z_{w\times h}\}$ - set of sites
$\mathbf{\Lambda} = \{\lambda_1, ...\lambda_l\}$ - state space
$x_0 = \{x_1, ...x_{w\times h}\}$ - initial label configuration for the sites in $\mathbf{Z}$
P - local conditional probability density function

Output:

$x = \{x_1, ...x_{w\times h}\}$ - output label configuration for the sites in $\mathbf{Z}$

Begin

Temperature initialisation
$t \leftarrow 1$
iter $\leftarrow 1$
$x \leftarrow x_0$
while iter $\leq$ Niter **do**
  for each site z_i **do**
    $\lambda \sim \mathbf{\Lambda}$ (random sample from the state space $\mathbf{\Lambda}$)
    $p \leftarrow \dfrac{P(\lambda, z_i, x)}{P(x_i, z_i, x)}$
    $q \sim [0, 1]$ (random sample from the interval $[0, 1]$)
    $p \leftarrow p \times t$
    if $p > q$ **then**
      $x_i \leftarrow \lambda$
    end if
  end for
  iter $\leftarrow$ iter + 1
  $t \leftarrow t \times k_{temp}$
end while

End

bel) is greater than the randomly sampled probability. By implementing the last two steps, negative probability jumps are accepted during the synthesis process. This means that, from time to time, proposals lowering the LCPDF at the current site with respect to the LCPDF of the current label are accepted. The purpose of this technique is to explore at best the state space in order to find the global maximum (and to avoid, thus, getting stuck in a local maximum).

As a single label value is tested for each site, one synthesis iteration (corresponding to a complete scan of the sites) via Metropolis algorithm is faster than via ICM. However, a more significant number of iterations is necessary for the Metropolis algorithm to converge.

Metropolis Simulated Annealing is an adaptation of the Metropolis algorithm inspired by the annealing process in metallurgy [Kirkpatrick 1983]. Likewise Metropolis, Metropolis SA is an iterative and stochastic relaxation technique belonging to the group of MCMC sampling techniques. The method relies on the introduction of a temperature term t. A high value of the temperature corresponds to a relaxation in the acceptance criterion for new labels propositions. Thus, bad trades are accepted in order to allow a wide exploration of the state space $\mathbf{\Lambda}$. The temperature is gradually

reduced by a temperature reduction coefficient k_{temp}, with $k_{temp} < 1$. As the temperature decreases with each iteration, the acceptance criterion becomes more restrictive, up to the point where only good trades (labels increasing the LCPDF) are allowed. The steps of the Metropolis SA method are given in Algorithm 2. The algorithm's steps correspond to Metropolis synthesis process in the special case when $k_{temp} = 1$.

4.3.1.3 LST Synthesis Steps

Learning Phase

The parametric MRF based LST synthesis approach at one scale of analysis is briefly explained in the following. The first step consists of computing the LE LST field $\mathbf{T}_{\text{in}}$ of the input texture $\mathbf{I}_{\text{in}}$. By choosing a value for the structure tensor's σ_T parameter, a scale of analysis is implicitly fixed. In a second step, for each pixel, all 3 component LE vector-form LSTs in the neighbourhood are concatenated to the LE vector-form LST of the central pixel (as previously described in Section 2.3.2.1 and illustrated by Figure 2.17 of Chapter 2). Thus, the extended input LE LST field is formed. The parameters of the statistical model chosen for describing the input extended LE LST field are further estimated. To this purpose, a multivariate Gaussian distribution or a mixture of multivariate Gaussian distributions of a priori defined number of mixture components, K, are employed. The aforementioned steps correspond to the learning phase of the method.

Synthesis Phase

The synthesis phase itself follows the learning phase. In a first step, the output LE LST field $\mathbf{T}_{\text{out}}$ is initialised with LE structure tensors randomly sampled from the input structure tensor field. The output LST field can have any size, it is not limited by the size of the input LST field. The synthesis of the output LE LST field is achieved pixel by pixel, in the order given by the a priori chosen image scan type. At each pixel, the LE extended structure tensor is composed, according to the considered neighbourhood configuration. A toroidal approach is considered for handling the borders of the rectangular grid on which the LST field is defined [Wei 2000]. In consequence, each point in $\mathbf{T}_{\text{out}}$ has a corresponding extended tensor. Given the positions x, y on a rectangular grid of size $w \times h$, the toroidal approach is implemented as follows: $\mathbf{T}_{\text{out}}(x, y, :) = \mathbf{T}_{\text{out}}(x \mod w, y \mod h, :)$ [1], where $\mod$ stands for the modulo operation and " : " denotes the 3 components of the LE mapped vector-form structure tensor. A decision step follows, where the current position is assigned a 3 component LE structure tensor from a set of possible entities $\mathbf{\Lambda}$, according to the rules of the relaxation algorithm employed. We remind here that the extended LE LST in one point is formed by the neighbouring LE LSTs and the LE LSTs of the current position. When a new proposition is made, a new state is tested uniquely for the LE LST corresponding to the current position. All the other LE vector-forms LST composing the extended LST in the current point (corresponding to the

[1] The modulo operation holds when considering the following indexing rule for the coordinates of the rectangular grid of size $[w \times h]$: $x = 0, ...w - 1$ and $y = 0, ...h - 1$.

neighbouring points) are left unchanged.

The decision criterion is based on the LCPDF. The LCPDF in each point is evaluated as the likelihood of the extended LE LST in the current position to the statistical model estimated on the input data. The extended LE LST is composed of the 3 component label proposition concatenated to the neighbouring LE LSTs. The procedure is iterated for several complete scans of the sites, until the predefined maximum number of iterations is reached.

The state space $\mathbf{\Lambda}$ comprises, in practice, all the vector-form LE LSTs values in the input field. This is true when Metropolis or Metropolis SA relaxation algorithms are employed. ICM algorithm does an exhaustive search of the optimum state in a site. Thus, for large sets of labels, the computational time for one site becomes very high, cumulating to a prohibitive computational time for several iterations of complete scans of the sites. To address this issue, a clustering on the original $\mathbf{\Lambda}$ set is applied. Thus, the set of labels $\mathbf{\Lambda}$, composed of all the LE LSTs in the input field, is replaced by a reduced set, composed of the class centers estimated by the clustering algorithm. To this purpose, K-means clustering algorithm is employed. While the computational time issues are solved, applying a clustering algorithm on the original state space results in a reduced accuracy. However, the experiments showed that this loss in accuracy does not have any negative consequence on the LST synthesis field, from the point of view of visual perception.

4.3.1.4 Neighbourhood Topology and Pixels Scan Type

The neighbourhood topology is often connected to the order in which the pixels are scanned during the synthesis process. From this point of view, two major neighbourhood topologies are identified, causal and non-causal.

All positions in a causal neighbourhood configuration correspond to sites for which the synthesis process at the current iteration of the relaxation algorithm has already been done. The shape of a causal neighbourhood is connected to the order in which the pixels are scanned. Raster scan order is a possible choice, where pixels are scanned line by line, from left to right. In this case, a causal neighbourhood configuration is in the shape of an L rotated by $\pi/2$ in the clockwise direction (see the half-square neighbourhood configuration in Figure 2.17 of Chapter 2).

Raster scan order coupled with a causal neighbourhood configuration is a popular choice when synthesising texture [Wei 2000]. As the synthesised information is propagated, pixel by pixel, few iterations of the relaxation process are needed for convergence. However, potential artefacts generated during the synthesis process will be propagated as well. In addition, for structured textures, a tendency to yield more regular outcomes has been observed, in the case of raster scan order and causal neighbourhood configurations [Urs 2012].

A random scan type is adopted in this work as well, during the synthesis process. In this case, a causal neighbourhood is not particularly needed. In consequence, a non-causal configuration is chosen. With a random scan order, the regularity and artefact propagation issues evoked for the case

of raster scan order are overcome. However, the synthesis process can be quite slow, as an important number of iterations are necessary for the relaxation algorithms to converge.

In addition to the type and size of the neighbourhood, the inter-pixel distance defines as well the neighbourhood topology. It corresponds to the distance between any two consecutive neighbouring pixels (see illustration in Figure 2.17 of Chapter 2).

4.3.2 Multiscale Extension

The monosclae LST synthesis approach described previously is extended to a multiscale approach. The multiscale synthesis method proposed in this work consists of a pyramidal method aiming at optimising the synthesis process. Several different pyramidal methods can be developed. In this work, we choose two methods among all possibilities for performing the multiscale synthesis. For a first proposed method, the synthesis process at a given scale is based solely on the information corresponding to the respective scale. For the second proposed method, the information at two scales of analysis is considered during the synthesis process.

4.3.2.1 Pyramidal Synthesis Principle

The synthesis process in one site relies on the information in its neighbouring sites. The size of the neighbourhood needs to be adapted to the size of the textural pattern, in order to fully characterise the underlying geometry of the input LE LST field. For LST fields composed of micro-patterns, neighbourhoods of small sizes are enough for characterising the geometry information. On the contrary, large size neighbourhoods need to be considered as the size of the LST field's composing pattern increases. Two issues arise when increasing the neighbourhood size. First, the dimension of the observed data (extended input LE LSTs) increases while the number of available sites for the learning phase diminishes, due to the border effect. This often leads to statistical inference problems, as previously described in Section 2.3.2.2 of Chapter 2. Second, the synthesis computational time increases.

To address the aforementioned issues, a pyramidal multiscale approach for LST field synthesis is proposed. The method consists of building a multiscale pyramid of the LST field. The lowest level of the pyramid, L_1, is associated to the highest resolution, namely to high frequency information (fine structural details). The highest pyramid level, L_s, corresponds to the lowest resolution, namely to low frequency information (macro structural information). The pyramidal synthesis principle is illustrated in Figure 4.3.

The input pyramid is built by computing an LE LST field per scale of analysis. This is achieved by fixing a value for the standard deviation of the Gaussian derivative kernel σ_G and tuning the standard deviation of the Gaussian weighting kernel σ_T for each of the s scales, with $\sigma_{T_1} < \sigma_{T_2} ... < \sigma_{T_s}$. Then, a decimation by a factor of 2 is applied in each dimension of the rectangular grid defining the sites of the LE LST field when passing from a level L_i to a level $L_{i+1}, i = 1, ..., s-1$.

A value σ_{T_1} is chosen for the standard deviation of the weighting kernel at the highest resolution pyramid level. Then, the standard deviation at all the lower resolution levels is computed as:

$$\sigma_{T_i} = \sqrt{\sigma_{T_1}^2 + (i-1)\alpha^2} \tag{4.3}$$

where α is a scale transition parameter. More details on the computation of the LST pyramid are given in Appendix E.

While the input pyramid is build from L_1 to L_s, the synthesis process starts with the highest level of the pyramid (lowest resolution), and continues, level by level, until reaching the lowest pyramid level L_1. When passing from a level L_i to its inferior level $L_{i-1}, i = 2, ..., s$, an up-sampling is performed by a factor of 2. More precisely, each LE LST in $\mathbf{T}_{\text{out}_\text{i}}$ is replicated on 4 positions in $\mathbf{T}_{\text{out}_{\text{i}-1}}$. The pyramidal approach consists thus in synthesising the main structural elements of the underlying geometry of a LE LST field and gradually adding more details while descending on the pyramid level.

4.3.2.2 Multiscale LST Field Synthesis with Monoscale Neighbourhoods

The main point of the multiscale synthesis approach based on monoscale LST field modelling is that the synthesis at one scale is based only on the information available at the respective scale. The only inter-scale transfer of information is done during the initialisation phase of all pyramid levels inferior to s. In this way, the synthesis at a level $i, i = 1, ..., s-1$ starts from a low resolution synthesis result, defining the main structural elements of the LE LST field. The principle consists of adding finer details to the main structural elements previously synthesised at the lower resolution pyramid level. However, the influence of the previously synthesised levels is marginal, as the structure can sometimes be completely lost after one complete scan of the LE LST positions.

The learning phase proceeds by computing an input LE LST field for each level of the pyramid. Second, according to the pre-defined neighbourhood topologies, the corresponding extended LE LST field is formed, at each scale of analysis. A different neighbourhood configuration can be defined for each scale of analysis. The parameters of the statistical models chosen for characterising the extended LST field at each pyramid level are further estimated. In the case of a multivariate Gaussian mixture model, the number of mixture components, K, is predefined.

The synthesis starts at the highest level of the pyramid, L_s. The output LE LST field $\mathbf{T}_{\text{out}_\text{s}}$ is initialised with randomly chosen LE structure tensors. Specifically, in each position, LE LSTs are randomly sampled from the input LE LST field at the scale s, $\mathbf{T}_{\text{in}_\text{s}}$. The synthesis at scale s proceeds. When a label proposition is made in a given point, the extended structure tensors in that point is build and its likelihood to the previously estimated model at the s^{th} pyramid level is evaluated. The point by point synthesis is reiterated until the convergence of the relaxation algorithm employed. The synthesis result obtained at level s is used for initialising the output LE LST field at the $s-1$ level of the pyramid. An up-sampling by a factor of 2 is also performed. The synthesis proceeds at level

Algorithm 3 Pyramidal LST synthesis with monoscale neighbourhoods

Input:

$\mathbf{I}$ - input image
s - number of scales
$\sigma_{T_i}, i = 1, ...s$ - structure tensor parameters
K - number of components of the multivariate Gaussian mixture model
$\mathbf{V}_i, i = 1, ...s$ - neighbourhood topologies
Niter - number of iterations of the relaxation algorithm

Output:

$\mathbf{T}_{\text{out\_1}}$

Begin

// Learning phase

$i \leftarrow 1$
while $i \leq s$ **do**
  compute input LE LST field $\mathbf{T}_{\text{in\_}i}$
  compose input extended LE LST field $\mathbf{T}_{\text{in\_ext\_}i}$ from $\mathbf{T}_{\text{in\_}i}$, according to $\mathbf{V}_i$
  estimate model M_i on $\mathbf{T}_{\text{in\_ext\_}i}$
  $i \leftarrow i + 1$
end while

// Synthesis phase

initialise $\mathbf{T}_{\text{out\_s}}$ with randomly sampled LSTs from $\mathbf{T}_{\text{in\_s}}$
$i \leftarrow s$
while $i \geq 1$ **do**
  iter $\leftarrow 1$
  while iter $\leq$ Niter **do**
    for each site z_j **do**
      build output extended LE LST $\mathbf{T}_{\text{out\_ext\_}i}(z_j)$ from $\mathbf{T}_{\text{out\_}i}$, according to $\mathbf{V}_i$
      $\mathbf{T}_{\text{out\_}i}(z_j) \leftarrow$ updated according to the rules of the employed relaxation algorithm applied on $\mathbf{T}_{\text{out\_ext\_}i}(z_j)$ based on M_i, with $\mathbf{\Lambda} = \mathbf{T}_{\text{in\_}i}$ (or quantified $\mathbf{T}_{\text{in\_}i}$ - for ICM)
    end for
    iter $\leftarrow$ iter + 1
  end while
  $\mathbf{T}_{\text{out\_}i-1} \leftarrow$ factor 2 upsampled $\mathbf{T}_{\text{out\_}i}$
  $i \leftarrow i - 1$
end while

End

$s - 1$. For a new label proposition, the likelihood of its corresponding extended structure tensor to the model estimated at the current level during the learning phase is evaluated. The aforementioned steps are applied for every pyramid level transition.

The learning and synthesis steps of the pyramidal approach with monoscale neighbourhoods are given in Algorithm 3. The scheme of this synthesis approach is illustrated in Figure 4.3.

4.3.2.3 Multiscale LST Field Synthesis with Biscale Neighbourhoods

Biscale neighbourhoods are considered in order to constrain the synthesis process at the current scale $i, i = 1, ..., s - 1$ by the previously synthesised low resolution LST field. The purpose of this

technique is to assign more weight to the lower resolution results during the synthesis process in order to maximise the resemblance of the synthesised LST field at level L_1 to the input LST field given as example. Also, faster convergence of the relaxation algorithms is expected.

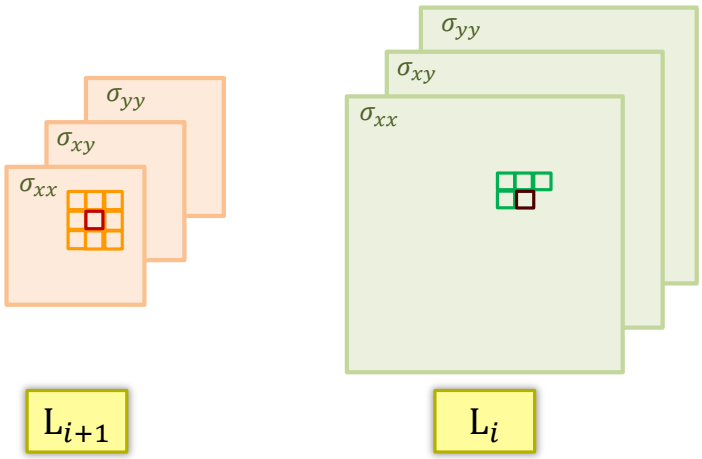


Figure 4.2: Example of biscale neighbourhoods for the pyramid level L_i and its neighbouring lowest resolution level L_{i+1}, with $i = 1, ...s - 1$, s being the number of pyramid levels.

During the learning phase, for all pyramid levels from 1 to $s - 1$, extended LE LST fields are built, comprising LE LST neighbourhoods at the current and the next pyramid level, as well. As for the highest pyramid level, s, the extended LE LST field is formed only by neighbourhoods belonging to the current scale. Different neighbourhood topologies can be adopted for each scale of analysis. For each pyramid level ranging from 1 to $s - 1$, two neighbourhood topologies are defined, one for the current scale and one for the previously synthesised scale. For the current scale, a causal neighbourhood topology is preferred. Given that the level above has already been synthesised, non-causal neighbourhoods can be employed. An example of biscale neighbourhoods is illustrated in Figure 4.2. For the pyramid levels ranging from 1 to $s - 1$, the parameters of the predefined statistical models will be estimated on biscale extended LE LST fields. On the highest pyramid level, the statistical model is applied on a monoscale extended LE LST field.

The same considerations hold for the synthesis phase as for the previous approach (Section 4.3.2.2), with few exceptions. For all scales going from 1 to $s-1$, extended LE LST fields comprising biscale neighbourhoods are built for the output field as well. In addition, when a label proposal is made, the likelihood of the extended LE LST in that point to the multiscale estimated model is evaluated. It should be noted that for biscale neighbourhoods, the LST field at the upper scale is applied an up-sampling with a factor of 2, so that there is a one to one correspondence for all positions at the two scales. The scheme of this approach is illustrated in Figure 4.3.

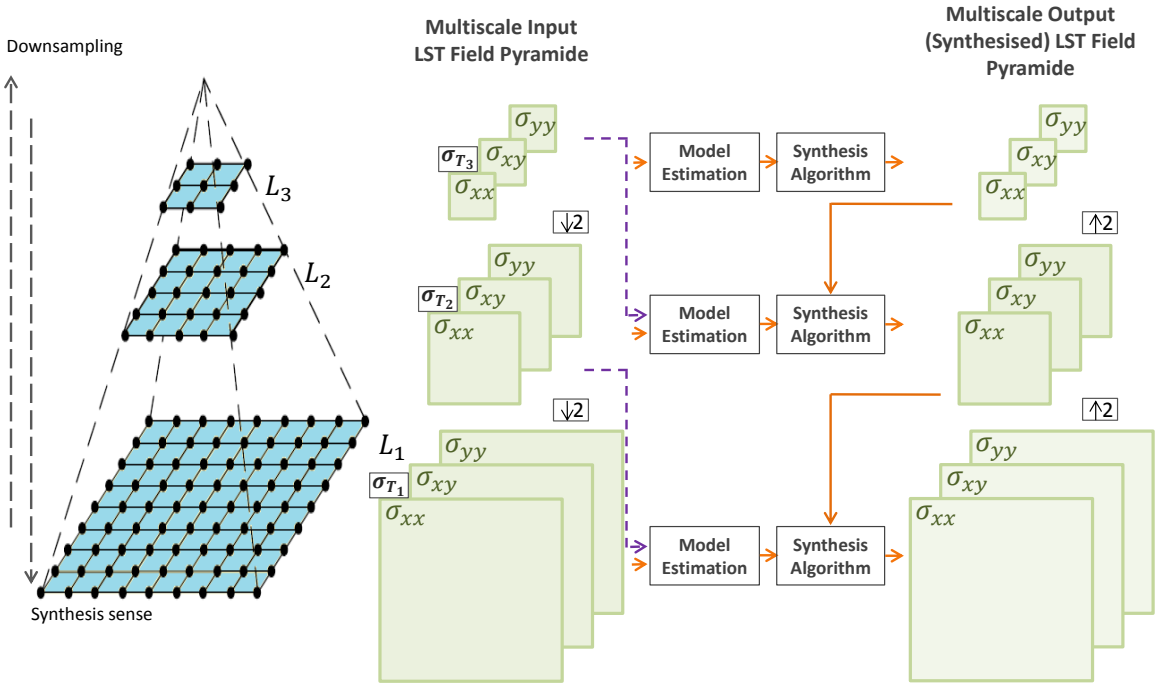


Figure 4.3: Multiscale LST field statistical modelling based synthesis scheme. Both monoscale and biscale neighbourhood approaches are illustrated. The following color code is used for the arrows connecting the scheme: orange for both approaches and purple only for the multiscale neighbourhoods approach.

4.3.3 LST Synthesis Algorithm Parameters

For all LE LST synthesis methods, a set of parameters need to be pre-defined by the user for launching the synthesis process.

In the case when the input LST field belongs to a real texture, the structure tensor parameters values (σ_G and σ_T) need to be first defined for computing the input LST field of the input texture. If a multiscale synthesis approach is employed, a value for σ_{T_i} has to be defined for each level of the pyramid, $i = 1, ...s$.

The number of mixture components of the statistical model used for characterising the LST field at each pyramid level needs to be defined as well. Different values can be chosen for each pyramid level. A neighbourhood topology associated to each pyramid level and the sites scan order are pre-defined as well.

For the synthesis process, a relaxation algorithm is chosen and the maximum number of iterations. Then, according to the algorithm, different parameters need to further be tuned. In the case of the ICM algorithm, as previously explained, the input LST field is reduced to a smaller set, for optimization purposes. K-means algorithm is used to this purpose and the number of clusters needs to be defined. The number of clusters corresponds to the size of the reduced input LST field representing the state space $\mathbf{\Lambda}$ that contains the new label proposals made during the synthesis process.

For a multiscale approach, a different size of the input LST field can be chosen for each level of the pyramid. For the Metropolis simulating annealing algorithm, the value of the temperature coefficient k_{temp}, used to iteratively cool down the temperature, needs to be chosen as well. The synthesis starts for a temperature value equal to one and it decreases, iteratively, at a rate given by the multiplicative coefficient $k_{temp} < 1$.

4.4 Results

The monoscale and multiscale synthesis approaches with monoscale neighbourhoods are first tested on artificially generated LST fields composed of structural patterns of different sizes. In addition to the aforementioned methods, the multiscale synthesis approach with biscale neighbourhoods is tested as well on LST fields computed on real textures. The orientation map is chosen in this section for representing the LST fields.

4.4.1 Toy Examples

4.4.1.1 Monoscale Approach - Sensitivity to Synthesis Parameters

All LST synthesis approaches rely upon a relatively large set of parameters that have to be predefined by the user. In order to test the synthesis outcome's sensitivity to the different parameters, ideally, optimal values should firstly be found. Then, successive testing could be done, where all parameters are fixed to their optimal value and then, one by one every parameter's impact is analysed by ranging its value and analysing its influence on the synthesis results. Naturally, comprehensive testing is difficult to achieve, the test scenarios that can be imagined being very vast.

A short analysis of this type is done in the following, for a very simple test scenario. It consists in generating 2 populations of artificial LE LST fields $\mathbf{T}_{\text{in\_a}_1}$ and $\mathbf{T}_{\text{in\_a}_2}$. The LE LST from the two groups are arranged on a rectangular grid in the form of a checkerboard composed of squares of size 2×2. The input field $\mathbf{T}_{\text{in}}$ consists, thus, of the two populations of artificially generated LE LSTs, following the pre-defined arrangement (see Figure 4.4). Extrapolated to textured data, this test scenario would correspond to the case of a highly-regular microtexture, composed of small size structuring patterns. This analysis has illustration purposes, and cannot be generalized to the case of LST fields of real textures, composed of different types of patterns and characterised, thus, by a higher complexity information. For this test scenario, the monoscale synthesis approach is employed.

The rectangular grids of the input and output LE LST fields are of size 128×128 and 48×48, respectively. Each LE LST population is built starting from pre-defined values of the LST parameters (λ_1, λ_2 and θ), as indicated in Table 4.1. A sample set of same size as the size of each LE LST population is generated for each parameter. A randomly generated Gaussian noise is added to each parameter sample set. The following values are chosen for the noise standard deviations: 0.5 for the

Table 4.1: Parameters of the two populations of artificially generated LE LST fields, $\mathbf{T}_{\text{in\_a}_1}$ and $\mathbf{T}_{\text{in\_a}_2}$

	$\boldsymbol{\lambda_1}$	$\boldsymbol{\lambda_2}$	$\boldsymbol{\theta}$
$\mathbf{T}_{\text{in\_a}_1}$	100	12	$\pi/3$
$\mathbf{T}_{\text{in\_a}_2}$	5	2	$2\pi/3$

eigenvalues sample set and 0.05 for the orientation sample sets. For each parameter triplet $(\lambda_1, \lambda_2, \theta)$, an LST if computed as: $\mathbf{RDR}^{\dagger}$, where $\mathbf{D}$ is the diagonal matrix composed of the eigenvalues λ_1 and λ_2 and $\mathbf{R}$ is the rotation matrix associated to θ. Next, each LST is mapped on the LE space and further represented as a 3 component vector form.

ICM Algorithm

For the ICM algorithm, the original state space $\mathbf{\Lambda}$ corresponding to the input LE LST field is reduced by means of clustering to 20 labels (from $16,384$).

A first analysis of the impact that the different parameters have on the synthesis results is done for a raster scan order and causal neighbourhood of inter-pixel distance equal to 1. The number of components of the LE multivariate Gaussian model is set to 8. After successive testing, 8 seems an optimal value for this parameter, for the size 2 square checkerboard LST configuration.

The impact of the neighbourhood size is assessed first. To this purpose, the size was varied from 3 up to 11 and for each case, the learning and synthesis phases have been repeated 10 times. During the synthesis phase, the ICM relaxation process is iterated 10 times. The results of this analysis are summarised in Table 4.2. The synthesis results are grouped into 4 types:

- *perfect*, when the checkerboard arrangement of the input LE LST field is reproduced with no defects during the synthesis process. An example of this type is illustrated in Figure 4.4;
- *original pattern with artefacts*, when the original arrangement is synthesised but there are marginal artefacts on the output LST field. Some examples of this type are illustrated in Figure 4.5. This phenomenon is common to the raster scan order. When a synthesis defect occurs, it is propagated during the relaxation process;
- *other patterns*, when the relaxation algorithm converges to structured solutions of different spatial arrangements than that of the input LE LST field. Some examples of this type are illustrated in Figure 4.6;
- *constant patterns*, when the output LST field is quasi-constant (the output LE LST field is entirely composed of only one of the two input populations).

Perfect synthesis outcomes are generated for all neighbourhood sizes different than 3. A neighbourhood of size 3 is too small for capturing the structure of the input LST field and the synthesis

Table 4.2: Synthesis results (occurrences out of 10 repetitions) according to the size of the neighbourhood. The results correspond to the ICM relaxation algorithm with causal neighbourhood and raster scan order.

	Neighbourhood size				
Synthesis result	**3**	**5**	**7**	**9**	**11**
Perfect	–	7	5	6	3
Original pattern with artefacts	–	3	4	3	–
Other patterns	10	–	1	–	3
Constant pattern	–	–	–	1	4

process has always converged towards a different spatial arrangement of the output. A causal neighbourhood of size 5 shows the highest number of positive occurrences. More precisely, 7 out of 10 tests yield perfect LE LST field synthesis. In addition, the other 3 outcomes are still converging towards the original pattern, but marginal artefacts are visible on the output patterns. For neighbourhoods of size superior to 5, a perfect synthesis is still achieved, with lower occurrences. However, the occurrence of perfect synthesis decreases as the size of the neighbourhood increases (> 9). In addition, for causal neighbourhood sizes larger than 5, the synthesis process becomes unstable (constant patterns or patterns different than the original one are sometimes synthesised). This instability increases as the size of the neighbourhood overpasses the optimal one. While too large neighbourhoods lead to unstable synthesis, too small ones are total deal breakers for the synthesis process.

Secondly, the neighbourhood size was set to 5 and successive testing has been performed for analysing the impact of the number of Gaussian mixture components, K on the synthesis process. The value of K has been ranged from 1 to 100. While positive outcomes have been observed for all values, for $K < 4$ the synthesis process is unstable. Some examples of synthesis illustrating this instability are given in Figure 4.7. In addition, as K increases, the estimation of the statistical model describing the LST field is slower and the relaxation process takes longer to converge. Values ranging from 8 to 15 are found as optimal in terms of both stability and computational expenses.

Generally, for a raster scan order and causal neighbourhood, the ICM algorithm converges in less than 10 iterations. For an optimal choice of the parameters ($K = 8, ...15$, and a neighbourhood size of 5 or 7), the relaxation process converges in $2 - 3$ iterations. The computation time for these cases is below 30 seconds [2].

The disadvantage of the raster scan order and causal neighbourhood is that when defects are produced, they are further propagated during the synthesis processe. This phenomeon associated to raster scan order has been previously observed in the case of texture synthesis, for different approaches [Wei 2000, Urs 2013, Akl 2016]. To overcome these issues, a random scan order can be employed during the synthesis process.

For a random scan order, there is no longer a justified reason behind using a causal pixel topo-

[2]The computation times given in this chapter correspond to MATLAB simulations, performed on a computer of processor Core i7 (3.50 GHz).

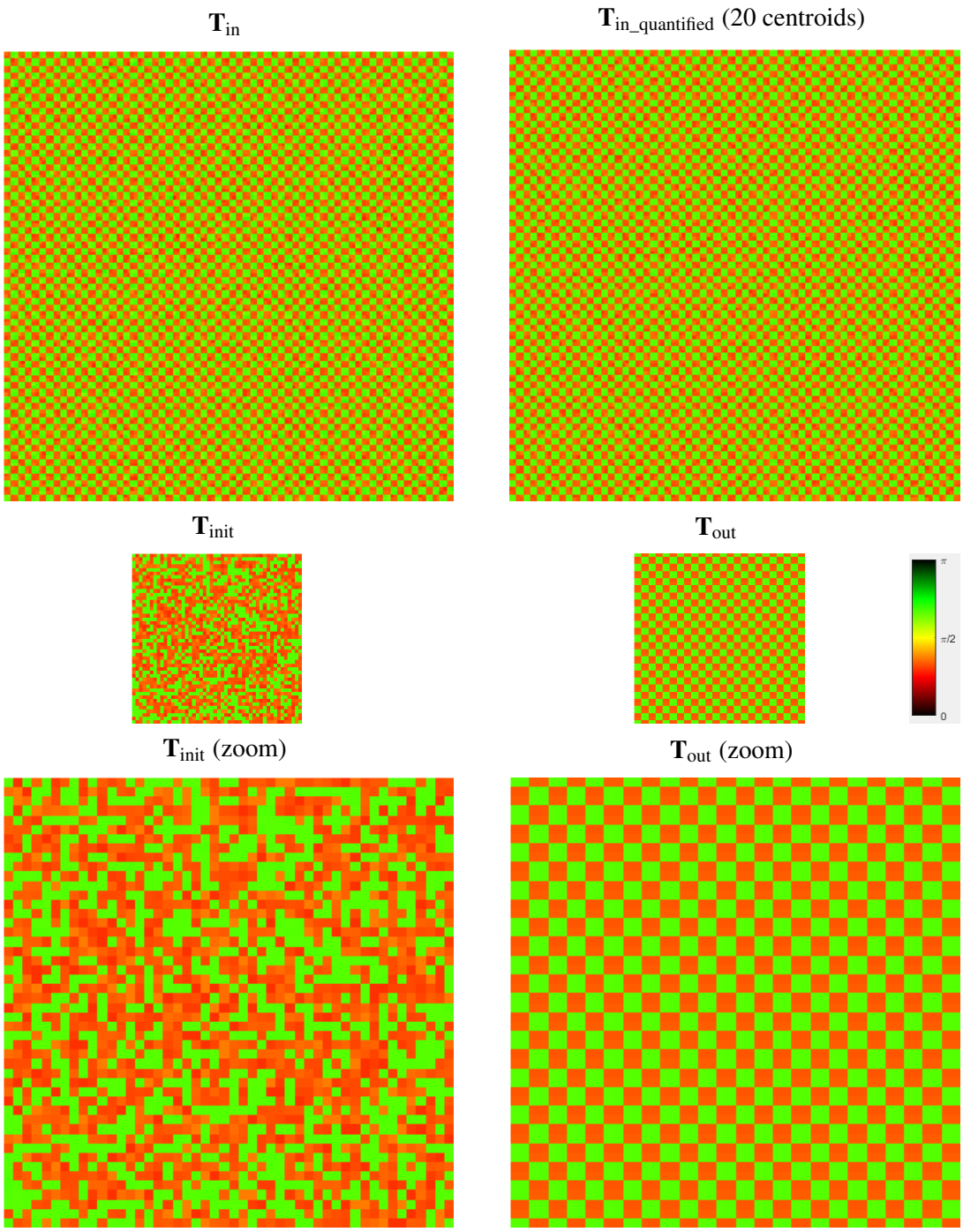


Figure 4.4: Example of a perfect LE LST field synthesis result. The original LST field $\mathbf{T}_{\text{in}}$ follows a checkerboard arrangement of size 128×128. The checkerboard squares are of size 2. $\mathbf{T}_{\text{in\_quantified}}$ corresponds to the state space $\mathbf{\Lambda}$ of the ICM algorithm, quantified to 20 states. The synthesised LST field of size 48×48 was obtained for a size 5 causal neighbourhood and raster scan order, after 3 iterations of the ICM relaxation algorithm. An 8 component multivariate Gaussian mixture model is employed.

logy. In consequence, cross and square neighbourhood topologies for an inter-pixel distance equal to 1 have been tested in this case.

The experimental synthesis results for the cross and square neighbourhood topologies are synt-

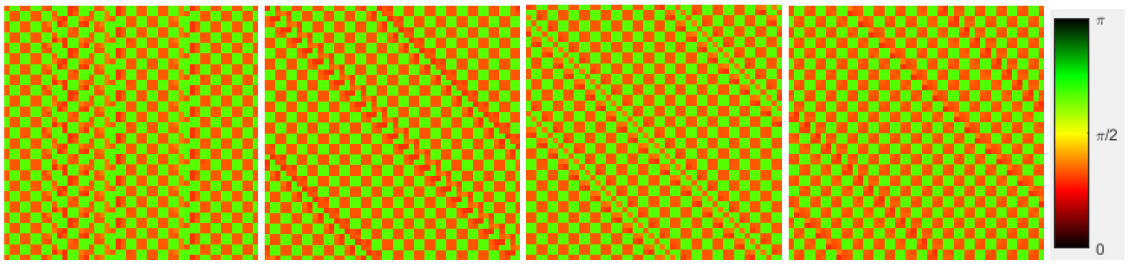


Figure 4.5: Examples of LE LST field synthesis results reproducing the original structure with artefacts. The results are obtained for causal neighbourhoods of size 5, raster scan order and less than 10 iterations of the ICM algorithm. An 8 component multivariate Gaussian mixture model is employed.

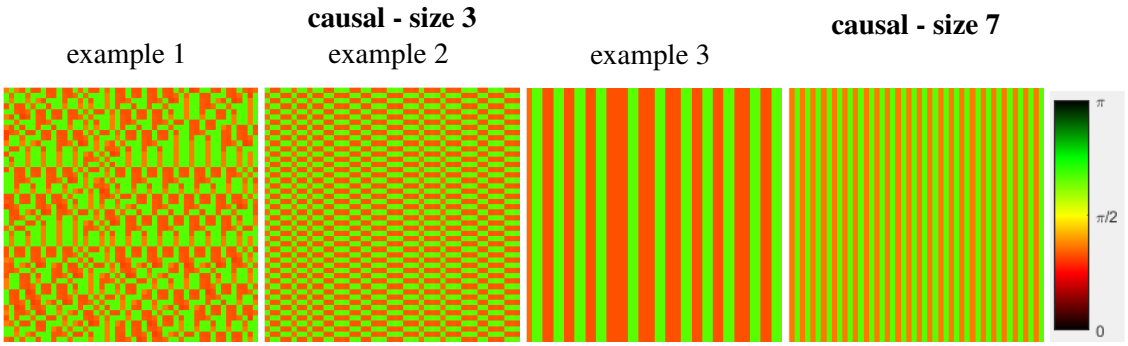


Figure 4.6: Examples of LE LST field synthesis results converging towards a structure different than the original one. The results are obtained for causal neighbourhoods of two sizes, raster scan order and less than 10 iterations of the ICM algorithm. An 8 component multivariate Gaussian mixture model is employed.

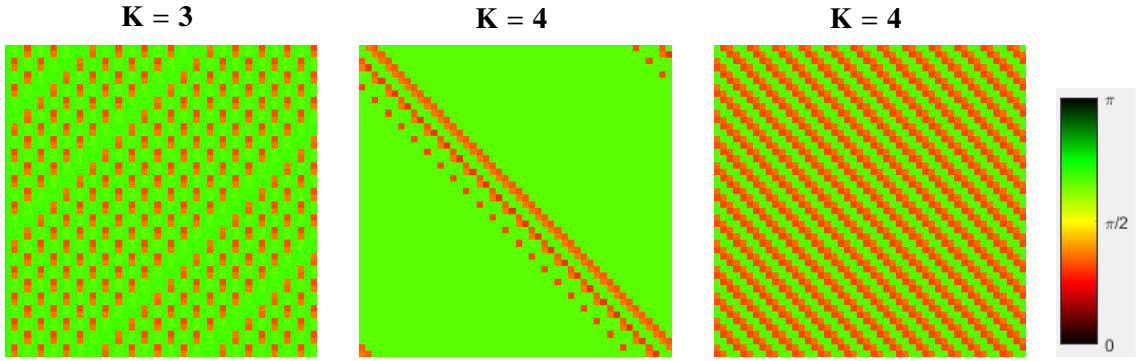


Figure 4.7: Examples of LE LST field synthesis results for a too small value of the number of Gaussian mixture components, K. The results are obtained for a causal neighbourhood of size 5, raster scan order and less than 10 iterations of the ICM algorithm.

hesised in the following. A size 3 square neighbourhood leads to unstructured results, a size 5 is optimal and a size 7 already leads to unstable synthesis results (patterns different than the original one, constant or unstructured outcomes). In contrast, for a cross neighbourhood topology, good and relatively stable results are observed for a size ranging from 9 to 13. As the size of the neighbourhood increases, the relaxation algorithm takes longer to converge. In addition, it has been observed that it is more difficult to break "bad patterns". More precisely, if most of the LST field is well

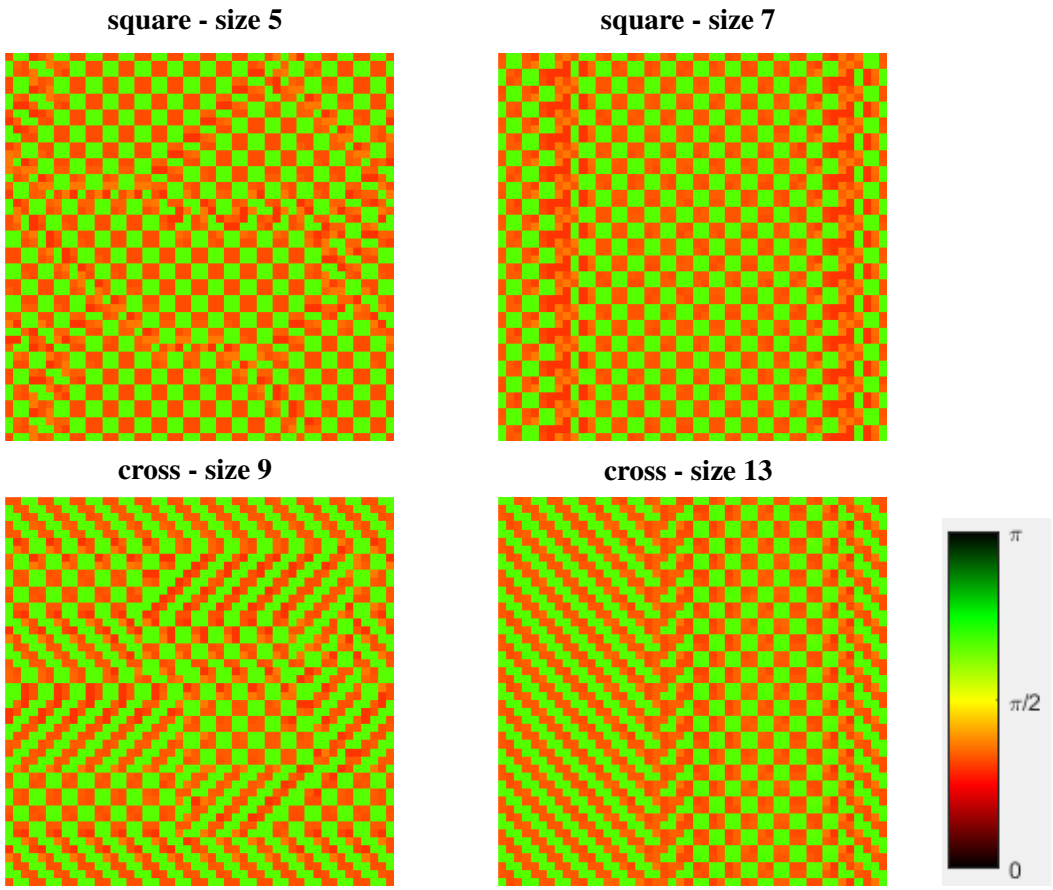


Figure 4.8: Examples of LE LST field synthesis results illustrating region merging artefacts. The results are obtained for random scan order and 100 iterations of the ICM algorithm. The results correspond to square and cross neighbourhood topologies of different sizes, as indicated in the title of each result. An 8 component multivariate Gaussian mixture model is employed.

synthesised, the small regions where patterns different than the original one have been synthesised take longer to be corrected or sometimes they are not corrected at all. In contrast, for smaller neighbourhoods, generally bad patterns are easily broken when the original pattern is synthesised over more than 50% of the output LST field (it takes only a few iterations until the original pattern is perfectly synthesised).

A cross neighbourhood topology associated to a random scan order provides a good compromise between the size of the data and the synthesis results. When increasing the neighbourhood size from p to $p+2$, and encompassing, thus, structural information from a larger area, there is an increase in the number of neighbours of only 4 points each time. In contrast, in the case of a square topology, when increasing the size of the neighbourhood configuration from p to $p+1$, $4p+2$ new positions are added to the neighbouring topology. We remind here that adding a point to the neighbourhood, corresponds to an order 3 increase in the size of the extended tensor (and, hence, in the dimension of the observation space). Thus, increasing the size of a square neighbourhood topology can easily lead to dimensionality problems and to stability issues during the statistical model estimation.

As opposed to the raster scan order when the ICM converges very fast ($2-3$ iterations), between

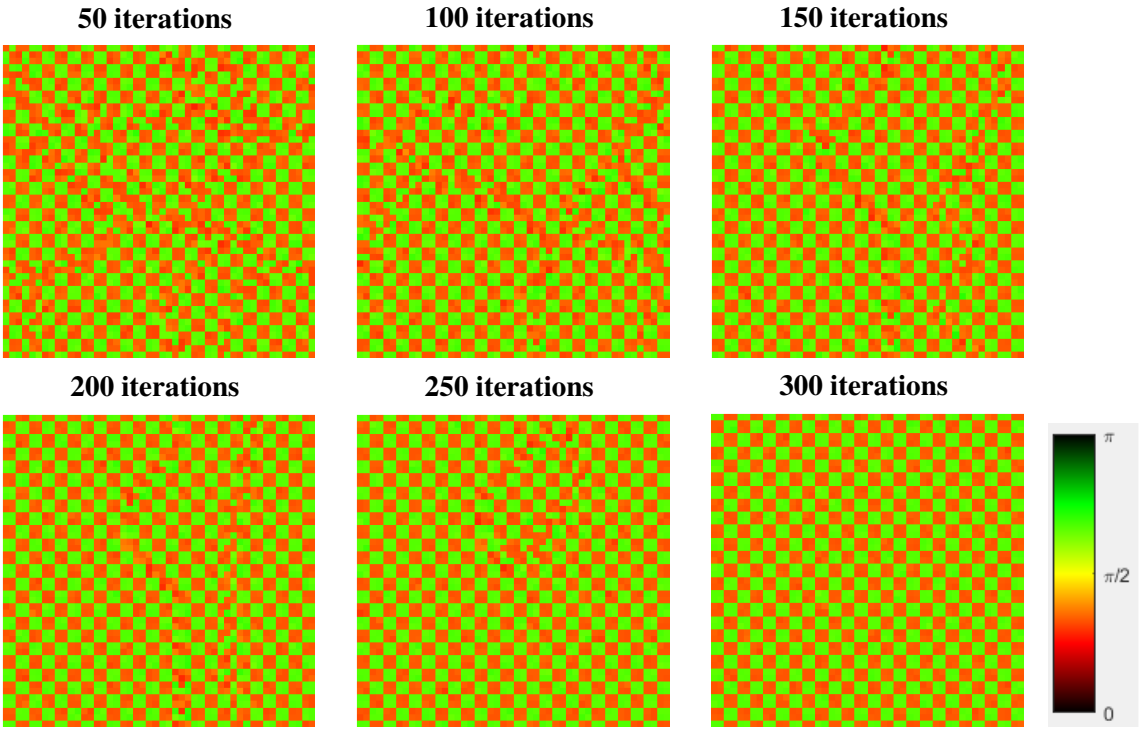


Figure 4.9: LE LST field synthesis results obtained after different number of iterations (as indicated on the title of each result) of the Metropolis SA algorithm. The results are obtained for a size 5 square neighbourhood topology and a random scan order. An 8 component multivariate Gaussian mixture model is employed.

50 and 100 iterations are generally needed for the ICM algorithm to converge when a random scan order is chosen. The computation time is of the order of few minutes, depending, naturally, on the size of the chosen neighbourhood configuration (e.g. 1 minute for a size 5 square neighbourhood and 50 iterations of the relaxation process).

In addition to higher computational time, there's another disadvantage of the random scan order associated with the ICM algorithms. In this case, the structure is synthesised by regions. Even if the structure of the input LST field is well synthesised on different regions, often there is a problem when merging two neighbouring regions. This phenomenon may occur, for instance, if the original pattern is well synthesised over 2 neighbouring regions but there is a shift by one pixel between the 2 patterns. This issue arises more often in the case of a square neighbourhood topology. To overcome this effect, a negative jump should be accepted for the relaxation algorithm for breaking the pattern and merging the neighbouring regions. As ICM algorithm does not allow negative jumps, the relaxation process gets stuck in a local maxima and the fusion defect cannot be corrected. Some other times, different patterns (original and parasite patterns) are synthesised over the different regions of the LST field. This issues arises more often in the case of a cross neighbourhood topology. Some examples of this kind are illustrated in Figure 4.8.

Metropolis and Metropolis Simulated Annealing Algorithms

Metropolis and Metropolis simulated annealing relaxation algorithms converge generally to the right solution and show stable behaviours for the parameter settings found as optimal, during the ICM successive testing analysis (raster scan order with size 5 causal neighbourhood, random scan order with size 5 square neighbourhood or size 9 to 13 cross neighbourhoods and 8 components in the Gaussian mixture model).

In the case of raster scan order and causal neighbourhood topology of size 5, the experiments showed that Metropolis algorithm needs generally over 200 iterations to converge. Metropolis SA accelerates the relaxation process by means of the temperature coefficient. Iteratively decreasing the temperature, the number of accepted bad trades is reduced over time and the algorithm converges faster, in consequence. For a raster scan order and size 5 causal neighbourhood and a value of the temperature coefficient equal to 0.99, Metropolis SA takes generally less than 100 iterations to converge. However, it can sometimes take up to 400 iterations. The associated computation times (including the learning and synthesis process) for 100 and 400 iteration of the relaxation algorithm are $\sim$ 5 minutes and $\sim$ 9, respectively. A lower value of the temperature coefficient leads to a faster convergence. However, as the solution exploration space is limited faster, the algorithm might converge toward a local maxima.

For a random scan order and a size 5 square neighbourhood topology, Metropolis SA algorithm takes generally between 300 and 400 iterations to converge (for a temperature coefficient equal to 0.99). The computation time for the learning and synthesis phase for 400 iterations of the relaxation algorithm is around 15. The advantage of Metropolis or Metropolis SA relaxation algorithms is that the region merging defects associated to a random scan order, previously observed in the case of the ICM algorithm (examples illustrated in Figure 4.5), are overcome. As these relaxation algorithms allow negative jumps, artefacts can be corrected. Synthesis results, at different phases of the Metropolis SA relaxation process are illustrated in Figure 4.9. It can be seen that for the first iterations of the algorithm, the same region merging defects are noticed on the synthesised field, as in the case of the ICM algorithm. However, as the relaxation process is reiterated, the merging artefacts are corrected, until a perfect synthesis is achieved (after almost 300 iterations). While Metropolis and Metropolis SA algorithms yield slightly better synthesis results, this comes at a quite high computational cost.

For the monoscale toy example considered in this section, the best compromise between the synthesis results and the computation time has been achieved by the ICM algorithm, with raster scan order and a size 5 causal neighbourhood. The synthesis converges very fast, generally after less than 5 iterations. The learning and synthesis phase for 10 iterations of the relaxation algorithm take around 17 seconds. It yields relatively stable solutions, out of all repetitions of the algorithm, the synthesis always led either to a perfect result, or to a partial reproduction of the original pattern with

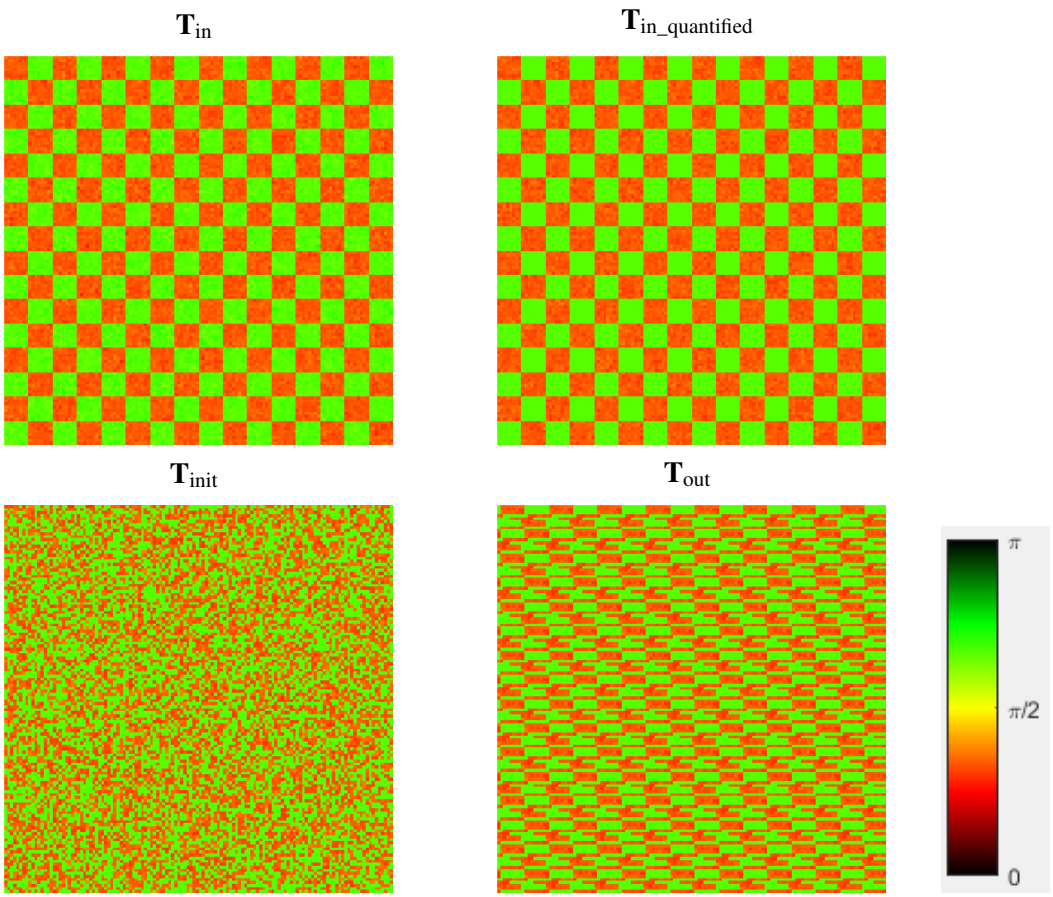


Figure 4.10: Example of monoscale LE LST field synthesis. The original LST field $\mathbf{T}_{\text{in}}$ follows a checkerboard arrangement of size 128×128. The checkerboard squares are of size 8. $\mathbf{T}_{\text{in\_quantified}}$ is the state space $\mathbf{\Lambda}$ of the ICM algorithm, quantified to 20 states. The synthesised LST field is obtained for raster scan order, causal neighbourhood of size 17, after 10 iterations of the ICM algorithm. An 8 component multivariate Gaussian mixture model is employed.

small artefacts. However, the raster scan order and causal neighbourhood topology propagate the potential artefacts arising during the synthesis process. A random scan type overcomes this issue, but the relaxation process takes longer to converge. In this case, artefacts might appear at the boundary between two synthesised regions. These artefacts cannot be corrected by ICM. This is because the ICM algorithm does not allow negative likelihood jumps and, in consequence, it often gets stuck in local maxima. These issues are overcome by Metropolis and Metropolis SA that, by accepting negative jumps in the likelihood, allow artefact correction. However, these advantages come at a very high computational expense. The computation time goes from a few seconds (necessary in the case of ICM), to several tens of minutes.

While the synthesis method becomes unstable for a wrong parametrisation (excessive size of the neighbourhood, a too small value for the number of components of the Gaussian mixture model), this toy example showed that the only deal breaker for a successful synthesis is a too small neighbourhood topology. More precisely, a size 3 neighbourhood proved to be too small to capture the structure of the input LE LST field, no matter the neighbourhood topology (causal, square or cross).

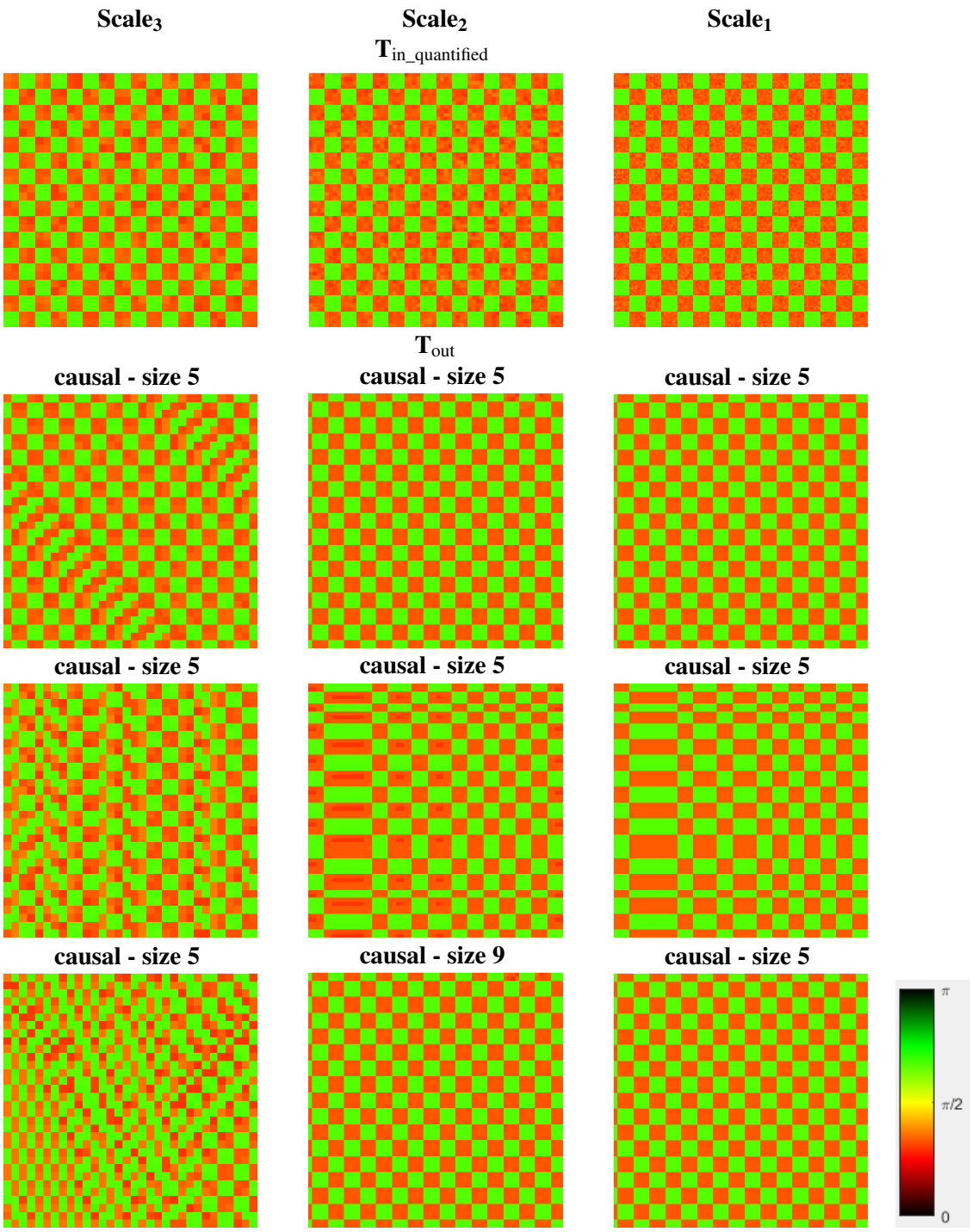


Figure 4.11: Examples of 3 scale pyramidal based LE LST field synthesis with monoscale neighbourhoods. The original LST field follows a checkerboard arrangement of size 128×128. The checkerboard squares are of size 8. $\mathbf{T}_{\text{in\_quantified}}$ is the state space $\mathbf{\Lambda}$ of the ICM algorithm, quantified to 20 for all scales. The synthesised LST field at each scale is obtained for raster scan order, after 5 iterations of the ICM algorithm. The neighbourhood topologies are given in the title of each result. A 6 component multivariate Gaussian mixture model is employed at each level of the pyramid.

4.4.1.2 Multiscale Approach

If the size of the input LST field's pattern increases, larger size neighbourhoods are necessary for a successful synthesis. However, increasing the size of the neighbourhood does not only yield higher

computational times, but may lead to statistical inference problems in response to the dimensionality increase of the observation space. For analysing this case, a second test scenario is created where the 2 populations of artificially generated structure tensors follow a checkerboard arrangement of squares of size 8.

In this case, the size of the pattern in equal to 16. A natural choice would thus be a size 17 causal neighbourhood. ICM algorithm is employed, with raster scan order. The rectangular grid of the input and output LST fields are of same size in this example, namely 128×128. The result obtained after 10 iterations are displayed in Figure 4.10. Not only that the synthesis process does not converge to the correct solution, but the computational time takes up to 19 minutes. A potential reason of the unsuccessful synthesis stands in the high dimension of the observation space induced by such a large neighbourhood. Considering the remarks made in Section 2.3.2.2 of Chapter 2, the dimensionality problem combined with limited-size input LST fields, can lead to model estimation issues that later manifest as the incapacity of synthesising the original pattern of the LST field.

The multiscale pyramidal synthesis approach based on monoscale neighbourhoods is tested as well in this case. This allows to gradually build the structure of the LST field, starting from a degraded resolution, up to the original one. In addition, when descending the pyramid levels, the synthesis at each level is supported by the previously synthesised lower resolution result. The rectangular grids of the input and output LST fields and the checkerboard squares associated to each scale of analysis are of the following sizes:

- L_3: 32×32 and 2 (lowest resolution);
- L_2: 64×64 and 4;
- L_1: 128×128 and 8 (highest resolution).

Some synthesis results obtained for a 3 level pyramidal approach are illustrated in Figure 4.11. In a first case, a size 5 neighbourhood topology has been considered for all scales of analysis. It should be noted that the LST field at L_3 corresponds to the same 2×2 squares checkerboard arrangement as in the example considered previously in Section 4.4.1.1. The experiments conducted in the previous section showed that in most cases, when a size 5 causal neighbourhood is employed, perfect synthesis results are achieved. However, in this case, most of the times the synthesis at L_3 is not perfect for a size 5 causal neighbourhood. This difference in results is explained by the size of the input data on which the statistical model is estimated. In the previous case, the rectangular grid of the input LST field is of size 128×128. Here, the LST filed at L_3 is 16 times smaller, as a result of the successive decimation applied when passing from high to low resolution pyramid levels. As a result of the decrease of the input data sample size, the model on L_3 is not estimated with a sufficient accuracy for providing a good synthesis of the input pattern.

If during the synthesis process at L_3 the original pattern is reproduced over most of the output LST field (as is the case of the first line of results illustrated in Figure 4.11), the original pattern

is perfectly reproduced up to the highest resolution. However, as the artefacts take more space on the synthesised LST field at lowest resolution, the synthesis at the higher resolution levels is compromised as well (see the second line of results illustrated in Figure 4.11). In such cases, if larger neighbourhoods are employed at L_2, the synthesis can be perfectly achieved, even when starting from a poor synthesis result obtained at L_3 (see the third line of results illustrated in Figure 4.11).

These results show that for a successful synthesis at the scale of origin, L_1, the synthesis does not need to converge to the correct pattern for all the lower-resolution scales. A smaller number of iterations can be considered at these levels, as a result. Moreover, by relying on the lower resolution synthesis result, smaller neighbourhoods than the actual size of the pattern can be employed, at most pyramid levels, without decreasing the synthesis quality. This is the case for all the examples in Figure 4.11, where a size 5 causal neighbourhood yields a perfect synthesis at scale 1, although the pattern size is 16. A consequence of reducing the number of iterations and the neighbourhood's size is a significant computational gain obtained for the pyramidal approaches. While the monoscale synthesis illustrated in Figure 4.10 was unsuccessful and took over 19 minutes, the multiscale synthesis illustrated in Figure 4.11 was successful and took less than 1 minute.

We emphasise here that the toy examples presented in this section illustrate very simple types of LST fields. In addition to being composed of only 2 populations, they are highly regular and of relatively small sizes. However, the LST fields of real textures are far more complex than that. The actual number of LST populations composing the pattern is in general considerably superior to 2. In addition, the spatial arrangement is more complex, less regular and composed of details defined at several scales of analysis. Furthermore, the patterns are substantially larger, requiring larger size neighbourhoods to be employed in order to fully describe the spatial arrangement of the input LST field. Nevertheless, as the complexity increases, more iterations are needed for the relaxation algorithms to converge, implying thus a higher computational time for each synthesis experiment. In consequence, the successive testing procedure necessary for finding the optimal values of parameters for one texture, becomes a burdensome task.

4.4.2 Real LST Fields

The synthesis of LE LST fields of two real textures is addressed in the following. A near-regular texture composed of a relatively simple pattern is tested first. Second, an anisotropic texture, less regular and composed of a more complex pattern is considered. Monoscale synthesis is tested, in addition to multiscale pyramidal approaches based on monoscale and biscale LST statistical models. For all approaches, ICM relaxation algorithm is employed and raster scan order associated to different types of neighbourhoods (causal and non-causal).

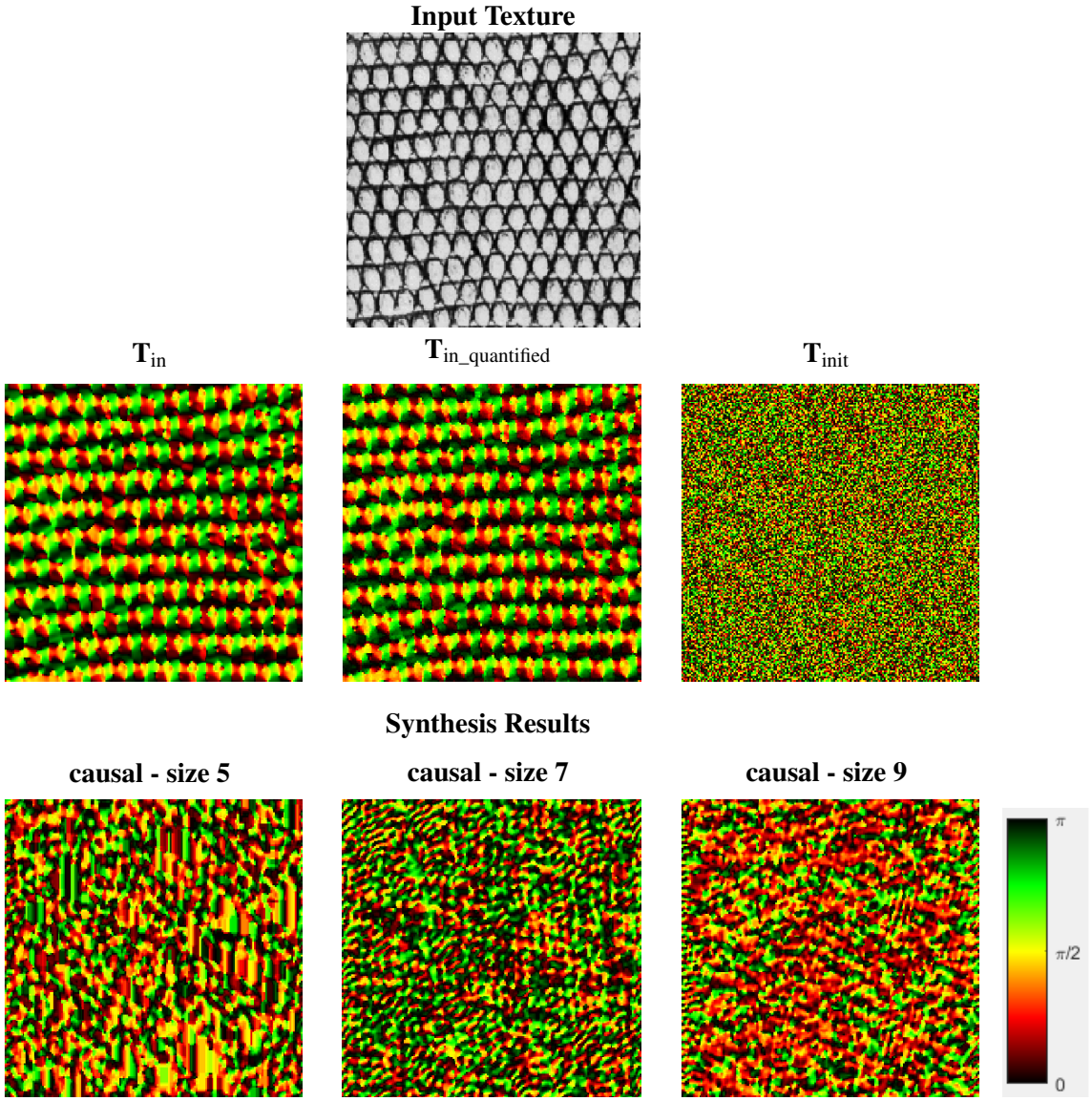


Figure 4.12: Examples of monoscale LE LST field synthesis of a Broadatz database texture extract. $\mathbf{T}_{\text{in\_quantified}}$ is the state space $\mathbf{\Lambda}$ of the ICM algorithm, quantified to 500 states. Different causal neighbourhood topologies are employed (given in the title of each result). The output LST fields are obtained for raster scan order, after 5 iterations of the ICM algorithm. An 8 component multivariate Gaussian mixture model is employed.

4.4.2.1 First Example

The first real LST field synthesis is performed on a highly structured texture corresponding to a 200×200 size extract of a Brodatz database texture.

Monoscale Synthesis

Figure 4.12 shows different monoscale synthesis results for the input LST field computed on the Brodatz database texture extract. The following parameter values have been considered for computing the LST field: $\sigma_G = 1$ and $\sigma_T = 1$. The results correspond to 5 iteration of the ICM algorithm. A raster scan order and several causal neighbourhood topologies have been considered. More preci-

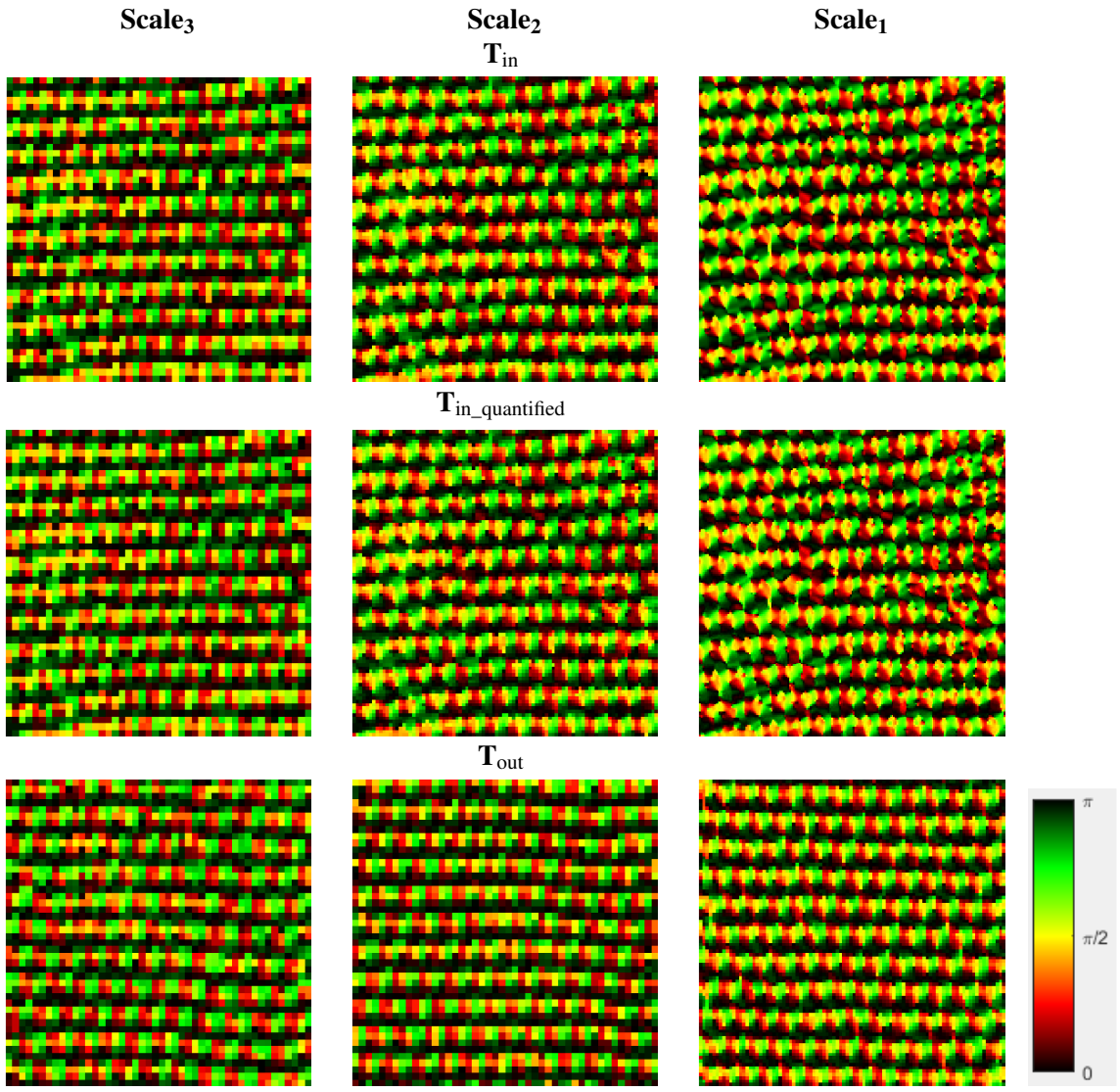


Figure 4.13: Pyramidal synthesis approach with monoscale neighbourhoods of the LE LST field of the Brodatz database texture extract illustrated in Figure 4.12. The values of the LST parameter σ_T at the different scales of analysis going from L_3 to L_1 are: $\sqrt{3}$, $\sqrt{2}$, 1. $\mathbf{T}_{\text{in\_quantified}}$ is the state space $\mathbf{\Lambda}$ of the ICM algorithm, quantified to 500, 1000 and 2000 labels, respectively, for the pyramid levels going from L_3 to L_1. The synthesised LST field at each scale is obtained for raster scan order, after 10 iterations of the ICM algorithm. Causal neighbourhoods have been employed: size 5, $d = 1$ (L_3), size 5, $d = 2$ (L_2), size 9, $d = 2$ (L_1). An 8 component multivariate Gaussian mixture model is employed at each level of the pyramid.

sely, neighbourhoods of sizes ranging from 5 to 19 have been tested, for two values of the inter-pixel distance, 1 and 2, respectively. Only the results corresponding to neighbourhood sizes going from 5 to 9 are illustrated here. A number of 8 components have been considered for the Gaussian mixture model. $\mathbf{T}_{\text{in\_quantified}}$ is the state space $\mathbf{\Lambda}$ of the ICM algorithm, quantified to 500 in this case.

For all the test scenarios considered, the monoscale LST field synthesis was unsuccessful. The inability to synthesise at one scale of analysis can be explained by the fact that neighbourhoods of sizes below the size of the textural pattern are too small and the texture cannot be correctly characterised. On the other hand, if neighbourhoods of sufficient sizes are employed, the dimension

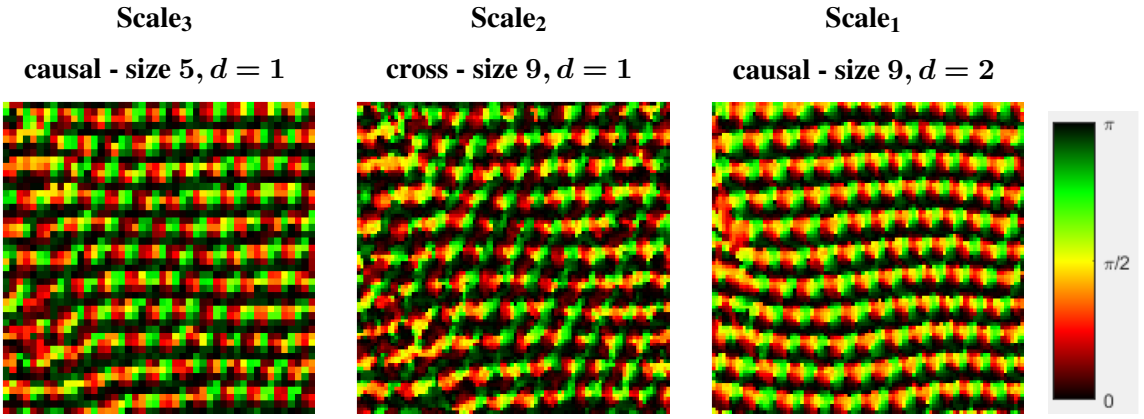


Figure 4.14: Representation of the output LE LST field for a 3 scale pyramidal based synthesis approach with monoscale neighbourhoods. The same Brodatz database texture extract, input, and quantified LST fields are employed as in the synthesis example illustrated in Figure 4.13. The synthesised LST field at each scale is obtained for raster scan order, after 5 iterations of the ICM algorithm. The neighbourhood topology for each scale is given in the title of each result. An 8 component multivariate Gaussian mixture model is employed at each level of the pyramid.

of the observation space becomes very large while the sample size diminishes, due to a larger border effect. In consequence, the estimated statistical model may provide a poor characterisation of the input LST field. These results highlight the difficulty of LST synthesis at the original scale of analysis and the necessity of employing a multiscale approach, in the case of real textures.

Multiscale Synthesis with Monoscale Neighbourhoods

A pyramidal approach on 3 levels is further employed, with monoscale neighbourhoods. The best two synthesis results obtained on this texture, as a result of successive testing are given in this section.

The following parameter values have been considered when computing the multiscale input LST field: $\sigma_G = 1$ and $\sigma_T = 1$, $\sqrt{2}$ and $\sqrt{3}$, respectively, for the analysis scales going from 1 to 3. The rectangular grids of the input and output fields (displayed in Figure 4.13) are of equal sizes, of following values, for the 3 levels of the pyramid:

- L_3: 46×46;
- L_2: 92×92;
- L_1: 184×184.

For each level of the pyramid, a different number of centroids was considered for quantifying the original state space, $\mathbf{\Lambda}$, namely $2000, 1000$ and 500 (from L_1 to L_3). As a higher resolution is characterised by finer variations in the structural details, a larger set of labels was considered as a more adapted choice in response to this richness of information.

The relaxation algorithm was iterated 10 times at each scale. Causal neighbourhood topologies were considered. Successive testing was performed in order to find the optimal values for the para-

meters. The experiments showed that a number of Gaussian mixture components above 6 should be employed in this case. A multivariate Gaussian mixture model of 8 components has been employed for characterising the LE LST field at each level of the pyramid.

The combination of neighbourhoods topologies that yields **the best LST synthesis results** in this case is the following one:

- L_3: causal size 5, $d = 1$;
- L_2: causal size 5, $d = 2$;
- L_1: causal size 9, $d = 2$.

where d represents the inter-pixel distance. For this parameter configuration, the LE LST synthesis takes around 55 minutes to complete. However, similar synthesis results are obtained for 5 iterations of the ICM algorithm. In this case, the computation time is reduced to approximately 28 minutes.

The LST field synthesis outcome that is closest in resemblance to its corresponding input LST field is obtained at the lowest resolution scale of analysis, L_3. For the second scale of analysis, although the previously synthesised pattern is well conserved, little details specific to this scale are added. This comes naturally, since the same neighbourhood is employed as for the previous scale, with an inter-pixel distance equal to 2. However, for different neighbourhood sizes or different values for the inter-pixel distance, the previously synthesised structural pattern is degraded. As for the highest resolution pyramid level, L_1, more details are added and smoother transitions are observed between the structuring elements of the composing pattern. The synthesis outcome is similar to the input LE LST field. However, the result is more regular than the input. In addition, while the global pattern of the input field is well synthesised, local orientation variations that are visible on the input LST field at L_1 are not reproduced.

An extremely high sensitivity of the LST synthesis process to the neighbourhood topology has been noticed. Not only the neighbourhood size has a critical impact on the synthesis result but the value of the inter-pixel distance, as well. Actually, any slight modification with respect to the causal neighbourhood topologies considered in the case illustrated in Figure 4.13, leads to unsuccessful synthesis results.

The second best synthesis result is obtained when considering for the pyramid level L_2 a non-causal neighbourhood topology. It consists of a size 9 cross neighbourhood, with inter-pixel distance equal to 1. For L_1 and L_2, the same neighbourhood topologies have been considered as in the case that yields the best synthesis result. The results are illustrated in Figure 4.14. As opposed to the previous example, since the inter-pixel distance equals one, more details specific to the second scale of analysis are added during the synthesis process at L_2. However, some local artefacts are synthesised as well. The final synthesis result at L_1 shows a higher level of regularity than the input LST field at this level. In addition, an undulation is observed on the synthesised pattern. This is a consequence of the artefacts synthesised at the lowest resolution level, L_3. They consist of a few

pixel shifts in the synthesised structure that show up as a stair-like effect (visible on the left-bottom side of the LST orientation map). This artefact is propagated during the synthesis process from L_3 to L_1. As L_1 is characterised by smoother transitions between the structuring elements and in response to the toroidal border management, the aforementioned artefacts appear as undulations on the synthesised LE LST field at this scale.

4.4.2.2 Second Example

A second real texture has been chosen for testing the LE LST field synthesis. It consists of an anisotropic texture of a carbonaceous material. It is characterised by lower level of regularity and more complex pattern than the previous texture. The considered image is of size 256×256.

Multiscale Synthesis with Biscale Neighbourhoods

A 3 scale pyramidal approach based on modelling biscale LST fields has been employed. Biscale neighbourhoods are considered in order to assign more weight to the previously synthesised low resolution scales, during the synthesis process, since synthesising the correct structure gets more difficult as the resolution increases. The following parameter values have been considered when computing the multiscale input LST field: $\sigma_G = 1$ and $\sigma_T = 2, \sqrt{5}, \sqrt{6}$, for the scale of analysis going from 1 to 3. The input and output LST fields are of equal sizes, at each level of the pyramid. The size of their respective rectangular grids has the following values:

- L_3: 59×59;
- L_2: 117×117;
- L_1: 234×234.

The original state space $\mathbf{\Lambda}$ is quantified to a different number of labels, for each level of the pyramid. More precisely, $1000, 500$ and 100 labels are considered for the scales ranging from 1 to 3. As in the case of the Brodatz texture and for the same considerations, broader state spaces are considered as the resolution increases. The input texture and the original and quantified LE LST fields at the 3 scales of analysis as well as the synthesis results are displayed in Figure 4.15. ICM relaxation algorithm has been iterated 5 times. Different combinations for the neighbourhood topologies have been tested. The neighbourhood topologies yielding the best LE LST synthesis results are as follows:

- L_3 - monoscale neighbourhood: causal size $9, d = 1$;
- L_2 - biscale neighbourhood: causal size $5, d = 1$ (current scale) and square size $5, d = 1$ (previously synthesised scale, L_3);
- L_1 - biscale neighbourhood: square size $3, d = 1$ (current scale) and square size $5, d = 1$ (previously synthesised scale, L_2).

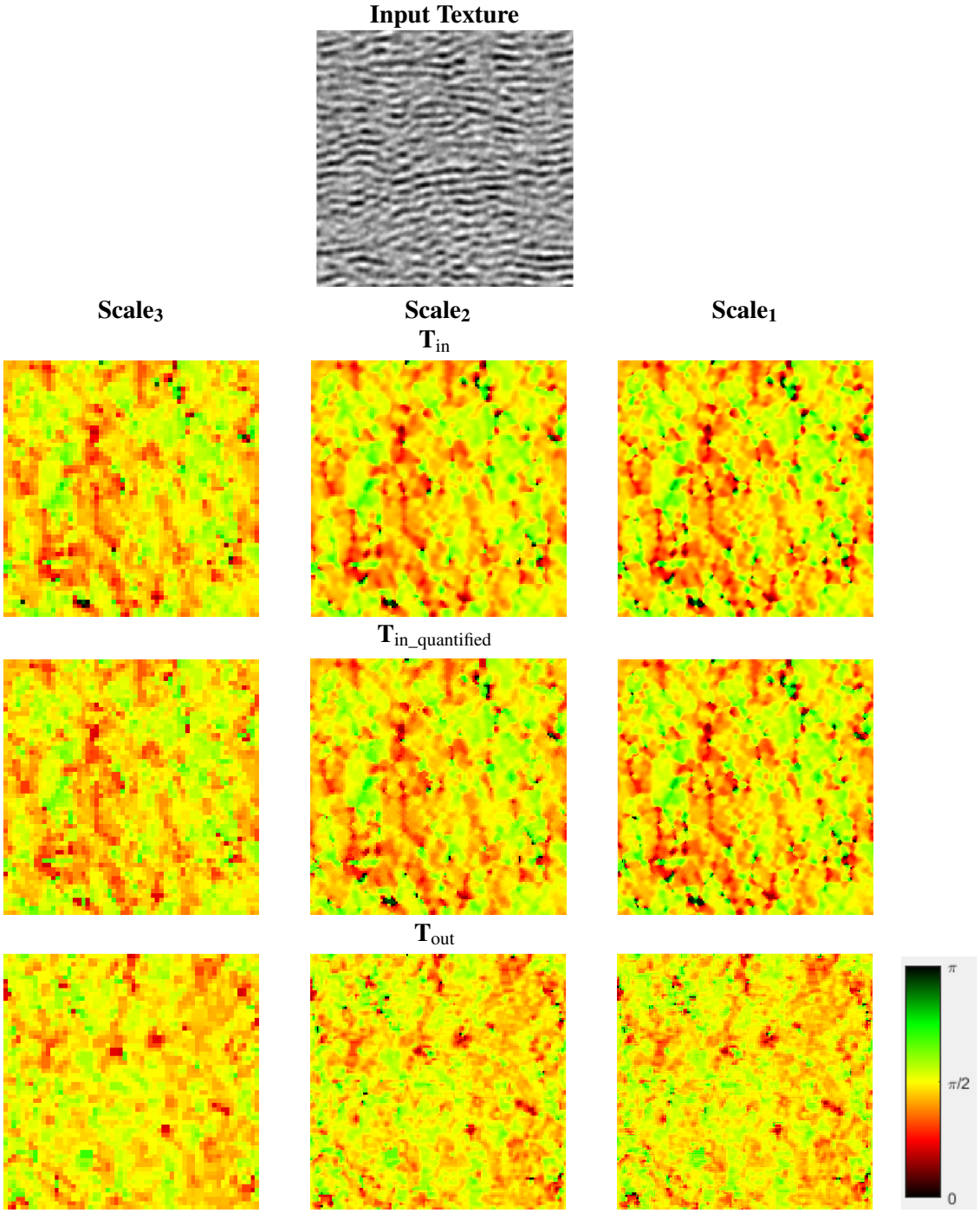


Figure 4.15: Pyramidal synthesis approach with biscale neighbourhoods of the LE LST field of a carbon material texture. The values of the LST parameter σ_T at the scales of analysis going from L_3 to L_1 are: $\sqrt{6}$, $\sqrt{5}, 2$. $\mathbf{T}_{\text{in\_quantified}}$ is the state space $\mathbf{\Lambda}$ of the ICM algorithm, quantified to $100, 500$ and 1000 labels, for the pyramid levels going from L_3 to L_1. The synthesised LST field at each scale is obtained for raster scan order, after 5 iterations of ICM. The following neighbourhoods are considered: L_3 - size 9 causal, L_2 - size 5 causal, L_1 - size 3 square and size 5 square (for the scales above L_1 and L_2). An 8 component multivariate Gaussian mixture model is employed at each scale.

A mixture model of 8 multivariate Gaussian distributions has been employed for describing the extended LST field at each pyramid level. The learning and synthesis phases, given the parameter configuration, take around 21 minutes.

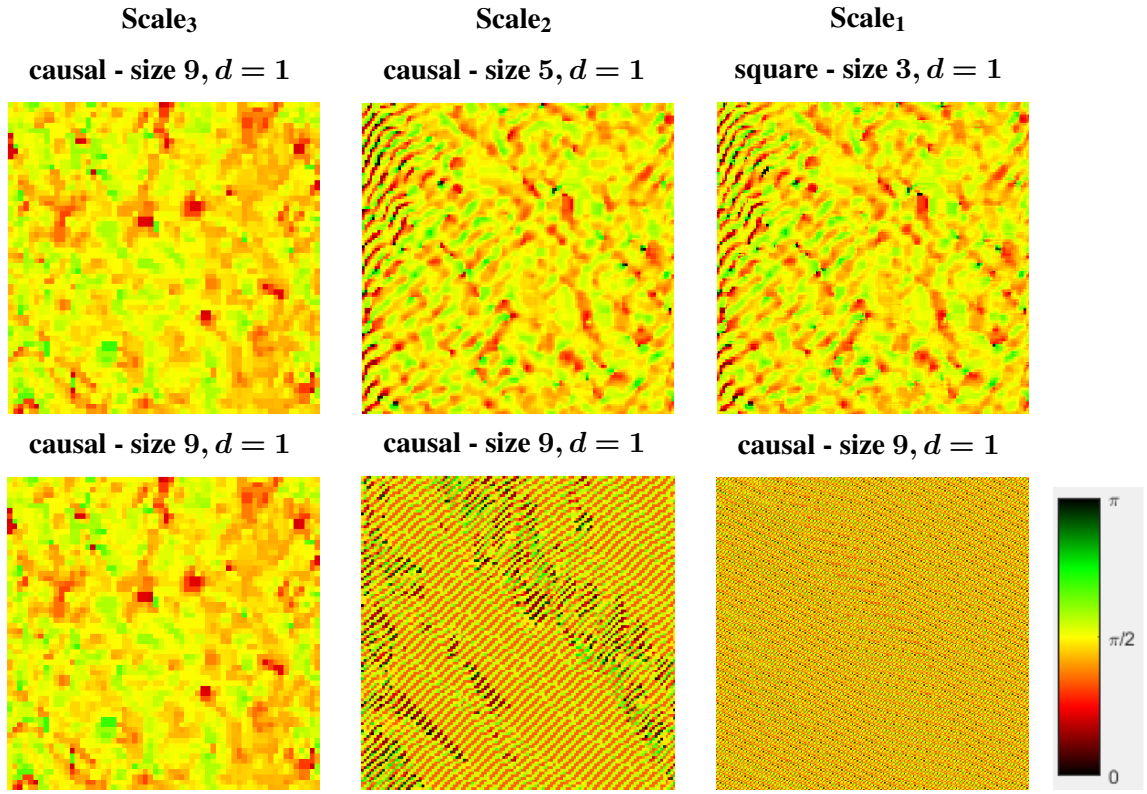


Figure 4.16: Two LST field synthesis results for a 3 scale pyramidal based synthesis approach with monoscale neighbourhoods. The same material example texture, input, and quantified LST fields are employed as in the synthesis example illustrated in Figure 4.15. The synthesised LST field at each scale is obtained for raster scan order, after 5 iterations of the ICM algorithm. The neighbourhood topology for each scale is given in the title of each result. An 8 component multivariate Gaussian mixture model is employed at each level of the pyramid.

The synthesised field at the lowest resolution pyramid level, L_3, is quite resemblant to its corresponding input LST field. As the resolution increases, it can be observed that the synthesised LE LST field is enriched in details. While this aspect is well marked between the pyramid levels L_3 and L_2, it is less pronounced between L_2 and L_1. In addition to adding finer details as the resolution increases, the structure defined at the lower resolution scale is well conserved, as en effect of the constraint imposed by employing biscale neighbourhoods.

As in the case of the Brodatz texture previously employed, a significant sensibility of the synthesis process to the size and shape of the neighbourhood has been noticed. For small variations in the size or shape of one of the neighbourhood topologies employed, the synthesis process degrades. The multiscale synthesis approach based on biscale neighbourhoods requires tuning not only one neighbourhood topology per scale, but a neighbourhood topology for the previous lower resolution scale as well, for all the pyramid levels from L_1 to L_{s-1}. In consequence, finding the optimal combination of neighbourhood topologies is quite demanding. It should be noted as well that if the set of values chosen for the σ_T parameter changes, the neighbourhood topologies yielding the best results in the previous case are no longer optimal.

Multiscale Synthesis with Monoscale Neighbourhoods

Figure 4.16 illustrates two synthesis results obtained for the same carbon texture, by applying a 3 scale pyramidal synthesis approach based on monoscale neighbourhoods. In the first case, at each scale of analysis, the same neighbourhood topologies has been considered as before (neglecting those defined for the previously synthesised scale). For this parametrisation, the LE LST synthesis starts degrading at the second scale of analysis. A stronger degradation of the output LST field starting from the second scale of analysis is observed for the second synthesis test, as well. In this case, a size 9 causal neighbourhood topology is employed at all scales. The synthesis degradation gets to the point that no structural component of the input LST field can be identified on the synthesised LE LST field at the highest resolution pyramid level, L_1.

The failure of multiscale approaches based on monoscale neighbourhoods to correctly synthesise, in these cases, the LE LST field up to the highest resolution, shows the benefits of employing biscale neighbourhoods. In this specific case, by employing biscale neighbourhoods, the correctly synthesised low resolution structure of the LST field is propagated up to the highest resolution, yielding, thus, a satisfying LE LST field synthesis result.

4.5 Conclusions and Future Work

In this chapter, the potential of the previously proposed LE statistical models has further been investigated for a task more complex than that of texture recognition. More precisely, their descriptive capacities have been analysed in the context of LST field synthesis. All the proposed approaches operate the synthesis on the LE space, but the output LE LST field can easily be mapped to the original tensor space by applying the matrix exponential to every LE LST in the synthesised field.

Probabilistic and parametric LST synthesis methods have been proposed, relying on a Markovian hypothesis. The synthesis is performed point by point, in an iterative manner. Several relaxation algorithms have been employed to this purpose. They consist of a deterministic algorithm, ICM, and of two stochastic algorithms, Metropolis and Metropolis SA, respectively.

Monoscale and multiscale pyramidal LST synthesis methods have been proposed as well. As far as the multiscale approaches are concerned, two possibilities have been developed. For a first multiscale approach, the synthesis at a given scale is based uniquely on the statistical models describing monoscale LST field spatial dependencies. For a second multiscale approach, the synthesis at a given scale relies on statistical models describing biscale LST field spatial dependencies.

All approaches have been tested on a small set of LST fields, either artificially generated or computed on real textures. Monoscale synthesis approaches worked well on the toy example composed of very small and simple structuring elements. However, the tests performed on real textures showed a significant difficulty to achieve the LST field synthesis at the scale of origin. This is because, generally, large neighbourhood topologies would be necessary for a correct characterisation of the input pattern. This would lead to an increase in the dimension of the observation space. In

addition, the size of the LST field on which the model needs to be estimated diminishes, since larger neighbourhoods give rise to larger border effects. In consequence, the statistical models are difficult to learn and the statistical inference process becomes unstable.

Multiscale approaches proved to be more adapted in these cases. In addition, for some textures, multiscale LST field synthesis based on statistically modelling biscale LST fields proved to be more adapted. By employing biscale models, more weight is assigned to the low resolution results that gradually support the synthesis process up to the highest resolution. Thus, new details are added at each level of the pyramid, while the previously synthesised structure is well preserved.

A main drawback of the proposed LST synthesis process is a very high sensitivity to the synthesis parameters and in particular to the neighbourhood topologies employed. In addition, the values chosen for the standard deviations of the Gaussian weighting kernels employed for computing the LST fields at the different scales and the neighbourhood topologies are interconnected. In consequence, for a given input texture, marginal changes on the LST field parameters will impact the neighbourhood topologies that need to be employed for obtaining a successful LST synthesis.

Neighbourhood tuning in the case of monoscale approaches is a task relatively easy to achieve, as the options are limited. However, this task becomes burdensome for multiscale models and even more delicate when biscale neighbourhoods are employed since there's a significant expansion in the possible combinations between the neighbourhood topologies chosen at each scale. In addition to neighbourhood tuning, the neighbourhood size can induce statistical inference issues. Monoscale synthesis methods for LST fields of large structural patterns are in particular prone to this phenomenon. In these situations large neighbourhoods should be employed that induce a significant increase in the dimensionality of the data. In consequence, the size of the input LST field might not be sufficient for an accurate estimation of the statistical model's parameters. The synthesis outcome is compromised as a result.

While the optimal neighbourhood topologies are particular to every texture and in some cases can take serious amounts of testing to find, we are positive that for every texture there is a combination of parameters leading to a satisfying LE LST field synthesis. While the synthesis itself is not so computationally expensive, finding the optimal set of parameters is.

The results obtained on both artificial and real texture LST fields show that the statistical models characterisation capacities go beyond the limit of LST field discrimination and up to the potential of LST field synthesis. While these first synthesis tests show promising results, the proposed methods definitely need further work, on different levels.

First of all, the statistical inference should be investigated more deeply on more complex artificially generated data in order to better understand the connection between the dimensionality of the input data and the size of the sample set that is necessary for an accurate estimation of the statistical model's parameters. Secondly, in order to address the dimensionality issues, solutions for reducing the dimensionality of the input data prior to model estimation can be imagined. In this case, the potential redundancy in the data may be eliminated and the associated statistical inference

issues may be solved, as a result. In addition, constraints can be imposed on the parameters of the statistical models, in order to reduce the number of degrees of freedom, such as imposing that the covariance matrix of the multivariate Gaussian model follows a particular pre-defined structure. In consequence, the covariance matrix could be expressed in terms of fewer parameters.

Another interesting prospect consists in developing a probabilistic but non-parametric approach based on Parzen windowing [Silverman 1986], as previosly done by several authors [Paget 1998, Urs 2013] in the case of texture synthesis. The method consists of assigning one Gaussian of predefined parameters to each data point in the input LST field. However, since very large sets of values would need to be tested during the synthesis process, this approach would be computationally expensive.

Conclusions and Future Work

"It always seems impossible until it's done."

— Nelson Mandela, Former President of South Africa

In this work, parametric and probabilistic approaches for modelling LST fields computed on textured images have been proposed. Since structure tensors are symmetric non-negative definite matrices that belong to a Riemannian manifold, the classical tools of the Euclidean geometry are not adapted for their characterisation. Two Riemannian statistical frameworks have been proposed in this work for modelling and discriminating LST fields. The respective frameworks are based on the AI and LE metric spaces.

On the AI space, a Riemannian Gaussian distribution is considered for modelling LST fields. The distribution is defined by a matrix-form center of mass and by a scalar dispersion parameter. Since the parameter estimates, in the sense of *Maximum Likelihood Estimation* method are not available in closed form, recursive estimation algorithms need to be employed, resulting in significant computational expenses. The LE mapping allows the representation of covariance matrices as vectors in the matrix logarithm domain. On such a vector space, a multivariate Gaussian model can be used for the description of LSTs. In this case, the parameter estimates are given in closed form. On both AI and LE metric spaces mixture models have been considered as well, in order to enrich the descriptive potential of the theoretical distributions. In these cases, the *Expectation Maximization* method has been employed for estimating the parameters of the statistical models.

Jeffrey's divergence has been considered as dissimilarity measure between statistical models. Its expression is not available in closed form, except from the case of multivariate Gaussian distributions. Solutions for its approximations based on Monte Carlo sampling techniques have been given for all the other cases.

Starting from equivalent values of the parameters, simulated covariance matrix sample sets have been drawn from the AI and LE statistical models. The randomly generated sample sets have been represented and analysed in order to evaluate and compare the descriptive potentials of the AI and LE statistical models. Justified by a higher number of degrees of freedom for a same order of the statistical model and confirmed by the experiments conducted on simulated data, the LE family of

statistical models showed higher characterisation capacities than those of the AI ones.

The descriptive capacities of the AI and LE statistical models has further been evaluated by analysing the fit of the theoretical distributions to empirical distributions of LSTs computed on textured images. To this purpose, a texture selection has been made comprising a wide diversity of textural patterns.

The methodology proposed on the AI metric space concerns the description of solely marginal distributions of structure tensors. On the LE metric space, the methods have been extended to model joint distributions allowing thus the characterisation of spatial and multiscale dependencies in an LST field. The spatial dependency information is extracted by considering neighbourhoods of LSTs. The multiscale information is expressed by LST fields computed for different values of the standard deviation of the Gaussian weighting kernel. Three joint approaches have been developed, two for modelling the spatial and multiscale dependencies alone and one for modelling them together.

Building extended structure tensors that encompass spatial and multiscale dependencies can easily be achieved. This task comes down to concatenating the three component vector-form LSTs corresponding to each point in the neighbourhood configuration or corresponding to the same point at the different scales of analysis. In contrast, statistically modelling empirical distributions of extended LSTs has proven to be a non-trivial task because of the significant increase in the dimension of the observation space that results in instability issues of the statistical inference process.

Three approaches have been proposed for the LST based methods to address the rotation invariance request that arises for a texture analysis method in many applications dealing with anisotropic textures. Two methods are based on estimating a texture's dominant orientation. However, since for some textures, the notion of global orientation is rather ambiguous, an exhaustive search approach has been proposed. It consists in searching the best orientation-wise match between the statistical models characterising the LST fields of two textures.

Both statistical frameworks proposed for LST modelling have been evaluated in the context of texture recognition. Two experimental frameworks have been developed, a content based image retrieval and a supervised classification protocol. The experiments have been conducted on very high resolution remote sensing images and on carbonaceous material images issued from high resolution transmission electron microscopy technology. While most data are textured, the methods are tested on non-textured data as well.

The experiments showed better classification performances for the LE statistical models over the AI ones. Not only that the classification accuracy is higher, but the LE methods are significantly faster. While on the AI metric space mixture models showed better recogition rates than one single Riemannian Gaussian distribution, on the LE space, generally, one multivariate Gaussian model yields the same retrieval rates as a mixture model. In addition, the algorithm is significantly faster when a single LE Gaussian model is employed. Moreover, on the LE metric space, joint LST distributions have also been modelled in order to describe the spatial dependencies in LST fields. The gain in performance with respect to the methods modelling marginal LST distributions is minimal.

The proposed methods have been compared against state of the art texture analysis methods as well. In all the conducted experiments, the LE statistical modelling approaches outperformed or yielded results of same order as the state of the art methods. In addition, the LST approaches showed to be particularly adapted to discriminate between texture classes encompassing significant differences in terms of anisotropy. In contrast, the state of the art methods showed to be less sensitive to the anisotropy level. In addition, the methods proposed in this work showed a constant behaviour on all datasets, as opposed to the methods used for comparison that, while performing well on some datasets, showed significantly lower performances on some other databases. These experiments proved as well the versatility of the LST statistical modelling based approaches that present a lower bias to the nature of the analysed textured data. Furthermore, when tested on datasets containing also non-textured data, the LST based methods outperformed all the other texture analysis methods previously tested on the same database.

The experiments conducted on texture recognition confirmed as well the intrinsic ability of LST based methods to handle rotation invariance, given the very high recognition rate obtained on the database composed of anisotropic textures of random orientations. In contrast, most of the state of the art methods provide a rather pseudo rotation invariance and in some cases, pre-processing of the texture database needs to be done before applying the methods.

The potential of the proposed statistical models for LST characterisation has further been assessed, beyond the original objective of texture recognition. More precisely, a Markovian hypothesis based methodology has been developed on the LE statistical framework for LST field synthesis. Monoscale and multiscale pyramidal synthesis approaches have been considered. Furthermore, two methods have been proposed for the multiscale approach. For a first method, the synthesis at a given scale of analysis is uniquely based on a statistical model characterising the LST field at the respective scale. In the case of the second method, the synthesis at a given scale is based on a model estimated on biscale LST fields. In this way, the synthesis results at low resolution scales are gradually supporting the synthesis process up to the original scale of analysis.

The first synthesis experiments performed on artificially generated LST fields showed that the monoscale synthesis methods are well adapted to the case of LST fields composed of small size structuring patterns. As the pattern size increases, larger neighbourhoods would be necessary to be employed for an accurate characterisation. As the dimension of the data increases fast when increasing the size of the neighbourhood configuration, large size neighbourhoods often lead to statistical inference problems. When it comes to LST field computed on real textures, multiscale synthesis approaches proved to be necessary, for all the textures considered in the experiments. For a good tuning of the parameters, successful LST synthesis results have been obtained.

A main drawback of the proposed methodology for LST field synthesis consists of a strong sensitivity to the synthesis parameters and in particular to the neighbourhood topologies employed. If for monoscale synthesis, neighbourhood tuning can be relatively easily achieved, for multiscale and especially for the approaches considering biscale neighbourhoods, finding the combination of

neighbourhood topologies adapted to each texture can be a quite burdensome task. However, in spite of these difficulties, we are positive that optimal parameter configurations leading to satisfying synthesis results can be found for any texture.

The results obtained so far with the LST statistical modelling based approaches have opened the way to some prospects for future work.

First of all, it can be interesting to explore more widely the choices of statistical models adapted for LST characterisation. A possible development can consist in employing a generalized Gaussian distribution for modelling LST fields on the LE metric space.

Another prospect consists in extending the LE LST to color. This can be achieved by concatenating the LE vector form LST computed at a pixel with the color information available at the respective pixel. In this way, the LE LST statistical modelling based approaches can be extended to characterise color textures [Abdelghafour 2018].

Improvements of the LST synthesis methods proposed in this work can also be imagined on several different levels. In a first step, a more thorough analysis can be conducted in order to evaluate the parameter estimation process when increasing the size of the observation space. Further experiments on simulated data could be performed in order to assess the relationship between the dimensionality of the observation space and the sample set size that would be necessary for a stable estimation of the statistical model's parameters.

A potential solution for avoiding the instability issues arising in large dimensional spaces consists in reducing the dimension of the extended LSTs prior to model estimation. In this way, redundancy in the data will be eliminated, too. Moreover, constraints could be imposed on the covariance matrix of the multivariate Gaussian model. The number of degrees of freedom of the statistical model may be reduced, as a result.

Probabilistic but non-parametric approaches based on Parzen windowing [Silverman 1986] can be imagined as well for LST field synthesis. Such an approach relies on assigning one Gaussian distribution of predefined parameters to every data point in the input LST field. While the statistical inference issues arising in some cases for the LST methods proposed in this work would be overcome, some other shortcomings would be faced. Since the input data would be characterised by a very large mixture of Gaussian distributions, evaluating the local conditional probability density function for each new label proposition made during the synthesis process would be quite computationally expensive.

Last but not least, the LST synthesis can be taken a step further and integrate its results in a texture synthesis framework. More precisely, similar to a the non-parametric approach previoulsy poposed in the litterature [Akl 2018], a parametric and probabilistic texture synthesis framework can be imagined were the texture synthesis is constrained by the structure layer synthesised by the LST field. To this purpose, a statistical model could be considered for characterising a texture's intensity values. Thus, new propositions for a pixel's value would be accepted or rejected on a criterion encompassing the structure tensor and intensity values likelihoods at that point.

Appendix A

Inter-class Diversity of VHR Remote Sensing Databases

A.1 Oyster Field Textured Patches Database

In this section, several textured patches examples belonging to each class of the VHR remote sensing oyster parks database are given (see Figure A.1). They are grouped in 3 classes: cultivated oyster racks, abandoned oyster fields and foreshore. The textured patches given as example in Figure A.1 illustrate the intra-class diversity specific to this database that is certainly challenging for the classification.

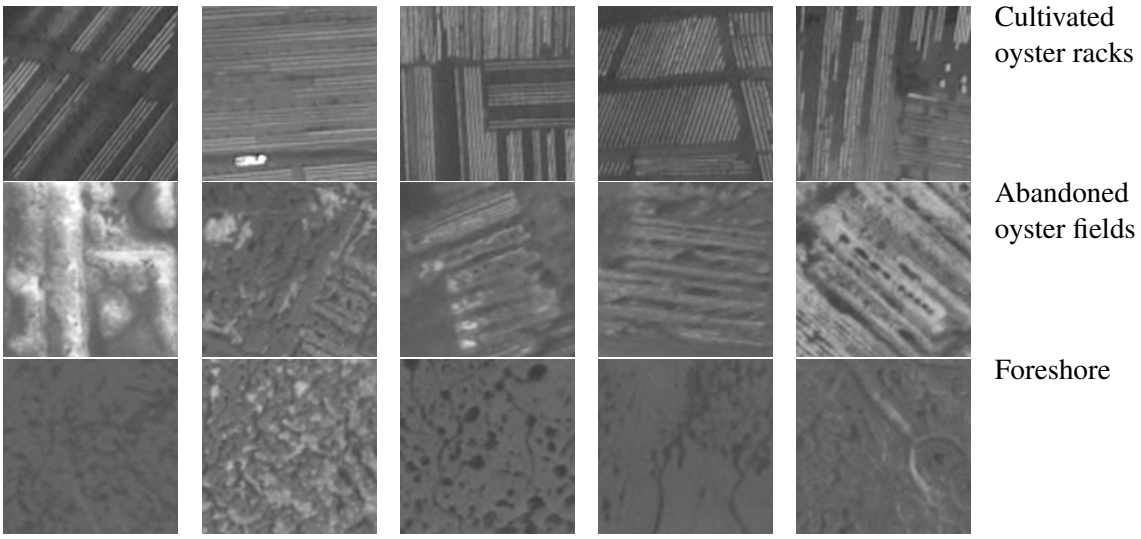


Figure A.1: Examples of oyster field textured patches from all 3 classes.

The first class is composed of cultivated oyster racks of lengths varying from several tens to a hundred meters. Several racks are generally organised together and the formed groups can be displaced at different distances and orientations relative to one another. Some of the oyster racks might be empty as well, all these aspects inducing a strong intra-class and even inter-patch diversity,

as illustrated by the textured patches given as example in Figure A.1.

As far as the textural information is concerned, the first two classes are composed of structured anisotropic patterns while the last class of foreshore is characterised by less structured textural patterns. This class is also more homogeneous.

A.2 Maritime Pine Forest Textured Patches Database

In Figure A.2 we illustrate several 256×256 textured patch extracts from each class of the VHR remote sensing maritime pine forest stands database. The forest stands are grouped in 4 age classes: $0-9$ years old, $10-19$ years old, >20 years old and a last class of clear cuts (0 years old).

For the first age class ($0-9$ years old), the tree ranks as well as the inter-rank forest floor and understory are visible on the image, due to a small diameter of the crown size (1 m) - see Figure A.2.

For the second forest class, the tree ranks are still visible, although the crown diameter increases, ranging from 1 to 3 m, resulting in a wider canopy. Thus, the textural content of this class is still anisotropic.

When it comes to class 3, the tree ranks are no longer visible on the image as the tree crowns are wider than 3 m. There is also a higher variety in terms of the density of population. Vegetation wholes might appear in some areas as a result of thinning operations and natural phenomena, too [Regniers 2014a]. In consequence, the textural pattern characterising this class is rather unstructured and less anisotropic. In addition, this class is characterised by a higher lever of intra-class diversity.

Finally, the fourth class of clear cuts consists of soil images after having removed all trees. However, some other small-size vegetation might still be visible on this areas. The textural pattern of this class is mostly isotropic.

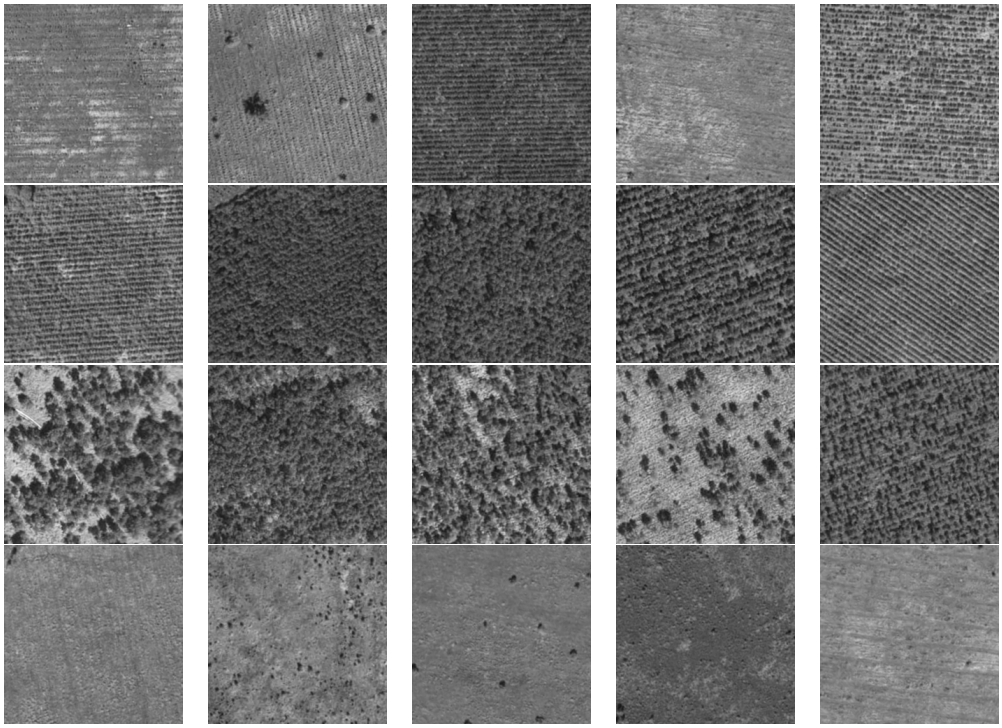


Figure A.2: Examples of maritime pine forest textured patches from all 4 age classes.

A.3 Land Use Land Cover Merced University Database

The UC Merced LULC database was formed by extracting homogeneous patches from USGS (United States Geological Survey) National Maps covering different US regions.

The database consists of both textured and non-textured LULC orthorectified aerial images grouped in 21 classes. It is composed of RGB images. This dataset is characterised by a significantly pronounced intra-class and inter-class diversity, as illustrated by the example patches of each class given in Figure A.3.

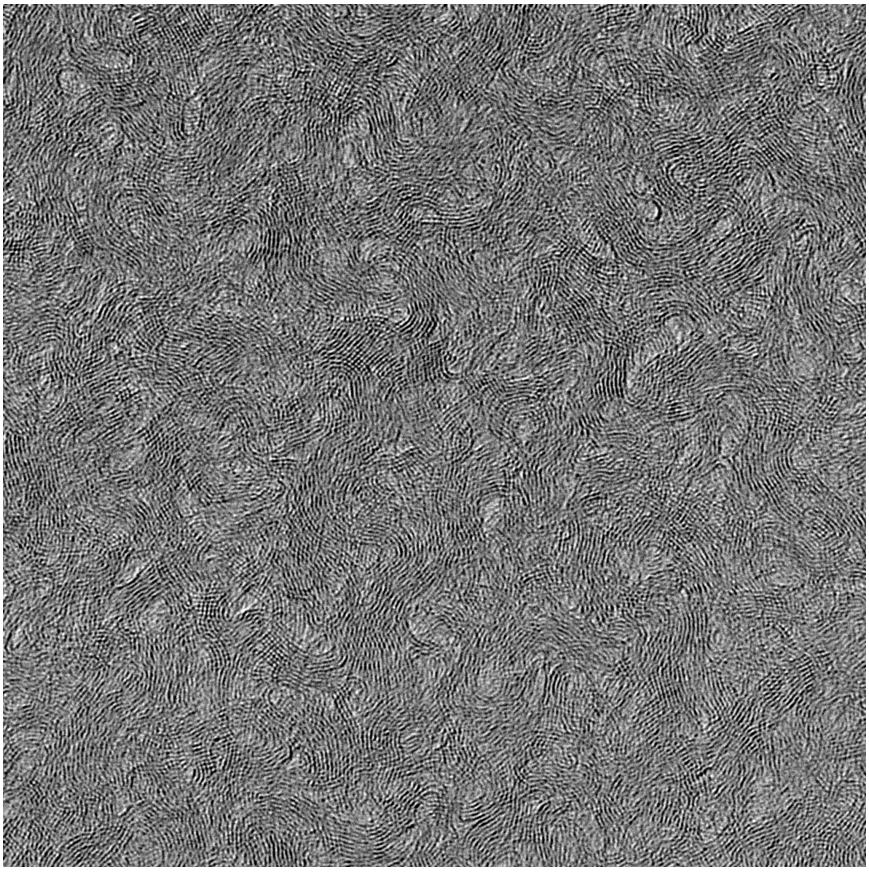

Forest
Freeway
Golfcourse
Harbor
Intersection
Medium residential
Mobile homepark
Overpass

Figure A.3: Examples of UC Merced LULC patches from all 21 aerial image classes.

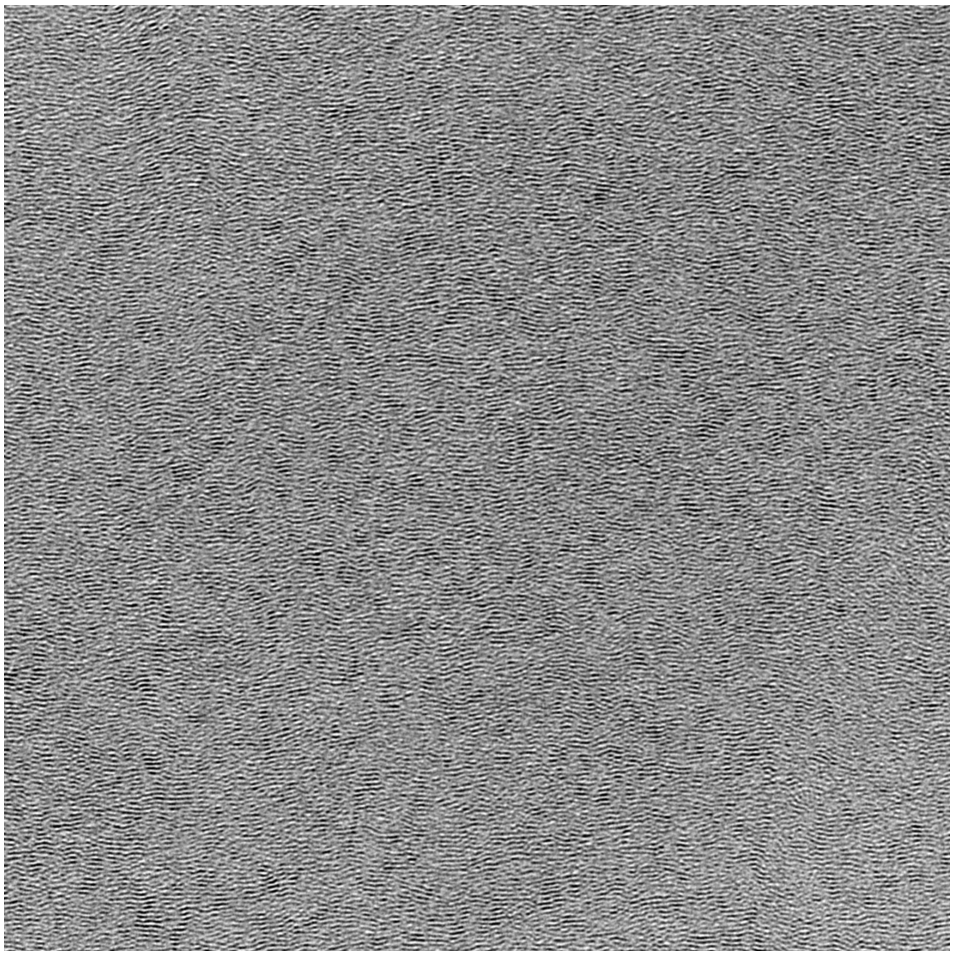
Appendix B

Original Snapshots of HRTEM Carbonaceous Material

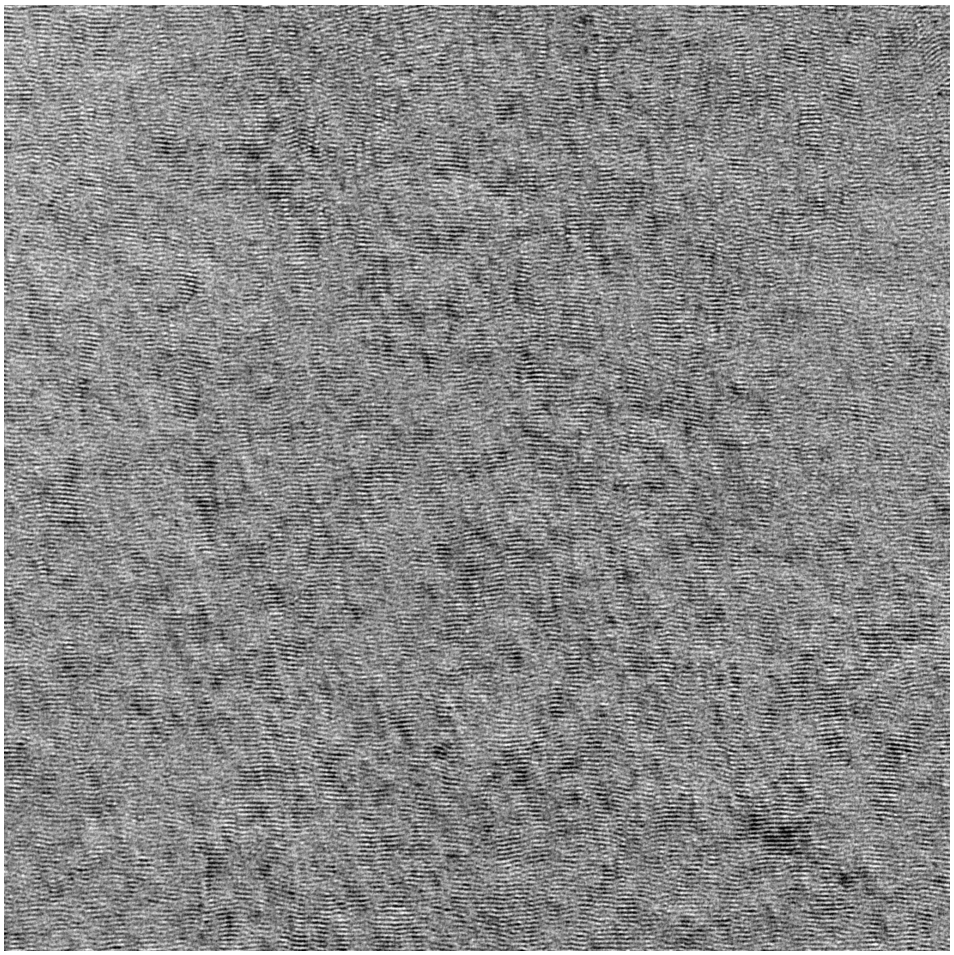
In this appendix we present the original HRTEM snapshots used for extracting the texture patches in the carbonaceous material database. Each snapshot of size 2048×2048 represents a different material and is used for extracting all the texture patches of a given class.



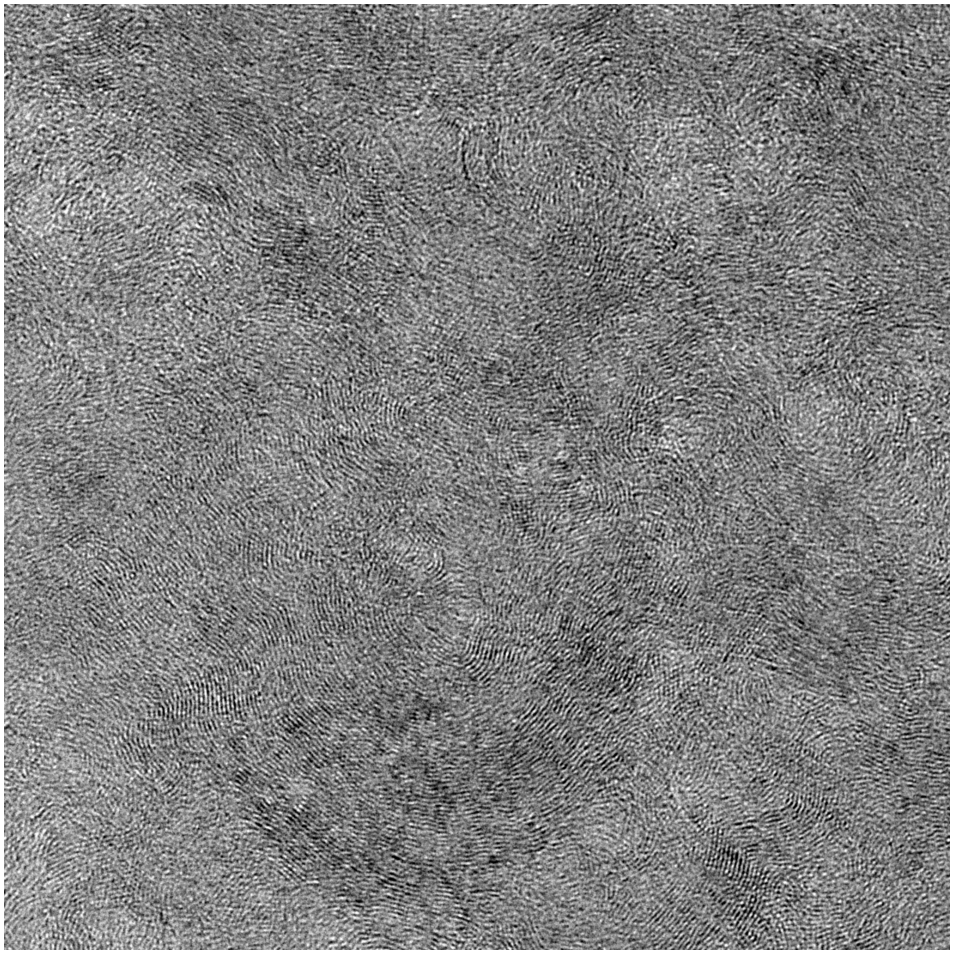
(a) PAN based carbon fiber



(b) regenerated laminar PyC



(c) rough laminar PyC



(d) smooth laminar PyC

Figure B.1: Original HRTEM snapshots of the 4 different types of carbonaceous materials

Appendix C

EM algorithm - Riemannian Gaussian Mixture Model

Let us consider a mixture model of K Riemannian Gaussian distributions of probability density function equal to the weighted sum of the probability densities of the K individual Riemannian Gaussian models, as given below:

$$p(\mathbf{Y}|(\omega_k, \mathbf{M}_k, \sigma_k)_{k=1,...,K}) = \sum_{k=1}^{K} \omega_k \, p(\mathbf{Y}|\mathbf{M}_k, \sigma_k). \tag{C.1}$$

The mixture model's parameters are: the centers of mass $\mathbf{M}_k$, the dispersions σ_k and the weights $\omega_k > 0$, of sum equal to one, associated to each distribution $k, k = 1, ...K$ in the mixture model.

Let $\mathcal{P}_m$ be the space of all $m \times m$ real matrices that are symmetric and strictly positive definite. Given a data set $\mathbf{Y} = \{\mathbf{Y}_1, ...\mathbf{Y}_N\}$, with $Y_n \in \mathcal{P}_m, n = 1, ...N$, drawn from a Riemannian Gaussian mixture model, the parameters are estimated by employing an adaptation of the *Expectation-Maximization* (EM) algorithm to the Riemannian geometry of the space $\mathcal{P}_m$, proposed by Said et al. in [Said 2017] and applied in [Rosu 2015a]. EM is an iterative algorithm that consists of two steps, the *expectation* step where for each data $\mathbf{Y}_n$ in the set, with $n = 1, ...N$, a responsibility score denoted by $\gamma_k(\mathbf{Y}_n)$ is assigned to each distribution $k, k = 1, ...K$ in the mixture model. This score indicates how much the Riemannian Gaussian distribution k is "responsible" for generating the data $\mathbf{Y}_n$. It is computed as the likelihood of the data sample $\mathbf{Y}_n$ to the k-th Gaussian distribution divided by the sum of the likelihoods of $\mathbf{Y}_n$ to all the K distributions in the mixture model. At the i-th iteration of the EM algorithm is holds:

$$\gamma_k^i(\mathbf{Y}_n) = \frac{\widehat{\omega}_k^{(i-1)} \hat{p}(\mathbf{Y}_n|\widehat{\mathbf{M}}_k^{(i-1)}, \widehat{\sigma}_k^{(i-1)})}{\sum_{j=1}^{K} \widehat{\omega}_j^{(i-1)} \hat{p}(\mathbf{Y}_n|\widehat{\mathbf{M}}_j^{(i-1)}, \widehat{\sigma}_j^{(i-1)})}. \tag{C.2}$$

The *maximisation* step of the EM algorithm follows, where the values of the estimates are re-computed given the scores obtained in the previous step. Thus, the parameters estimates of the

component $k, k = 1, ...K$ of the Riemannian Gaussian mixture model at iteration i are given by [Said 2017]:

$$\widehat{\mathbf{M}}_k^i = \arg\min_{\mathbf{M}} \sum_{n=1}^{N} \gamma_k^i d^2(\mathbf{M}, \mathbf{Y}_n) \tag{C.3}$$

where $d(.)$ stands for the geodesic distance defined in equation 1.17 of Chapter 1;

$$\widehat{\sigma}_k^i = \mathbf{\Phi}\left(N_k^{-1} \times \sum_{n=1}^{N} \gamma_k^i(\mathbf{Y}_n) d^2(\widehat{\mathbf{M}}_k^i, \mathbf{Y}_n)\right), \text{where } N_k = \sum_{n=1}^{N} \gamma_k^i(\mathbf{Y}_n) \tag{C.4}$$

and $\mathbf{\Phi}$ stands for the inverse function of $\sigma \rightarrow \sigma^3 \times \frac{d}{d\sigma} \log Z(\sigma)$. For more details, see [Said 2017];

$$\widehat{\omega}_k^i = \frac{N_k}{N}. \tag{C.5}$$

These steps are reiterated until a given convergence criterion or a number of maximum iterations is attained. The steps of the EM method for the estimation of the parameters of a Riemannian Gaussian mixture model are given in the Algorithm 4.

Algorithm 4 *Expectation-Maximisation* algorithm for a Riemannian Gaussian mixture model

Input: $\mathbf{Y}_1, ...\mathbf{Y}_N, K, Niter$
Output: $\widehat{\mathbf{M}}_k, \widehat{\sigma}_k, \widehat{\omega}_k, k = 1, ...K$
Begin
for $k = 1 : K$ **do**
  Initialise ω_k with $\frac{1}{K}$
  Initialise $\mathbf{M}_k$ with a random data sample from the input set $\mathbf{Y}_1, ...\mathbf{Y}_N$
  Initialise σ_k according to equation C.4, for $N_k = N$ and $\gamma_k(\mathbf{Y}_n) = 1$
end for
$iter = 1$
while ($iter <= Niter$) or (no convergence) **do**
  for $k = 1 : K$ **do**
    Estimate $\widehat{\mathbf{M}}_k$ according to equation C.3
    Estimate $\widehat{\sigma}_k$ according to equation C.4
    Estimate $\widehat{\omega}_k$ according to equation C.5
    $iter = iter + 1$;
  end for
end while
End

Appendix D

EM algorithm - Multivariate Gaussian Mixture Model

Let us consider a mixture model of K multivariate Gaussian distributions of probability density function equal to the weighted sum of the probability densities of the K multivariate Gaussian models, as given below:

$$p(\mathbf{Y}_{\mathrm{LE}}|\omega_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)_{k=1,...,K} = \sum_{k=1}^{K} \omega_k p(\mathbf{Y}_{\mathrm{LE}}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k). \tag{D.1}$$

The mixture model's parameters are: the means $\boldsymbol{\mu}_k$ and the covariance matrices $\boldsymbol{\Sigma}_k$ of the K multivariate Gaussian distributions and the set of mixture weights $\omega_k > 0$, associated to each distribution, with $\sum_{k=1}^{K} \omega_k = 1$.

Given a data set $\mathbf{Y}_{\mathrm{LE}} = \{\mathbf{Y}_{\mathrm{LE}_1}, ...\mathbf{Y}_{\mathrm{LE}_N}\}$, drawn from a multivariate Gaussian mixture model of size D, the parameters are estimated using the *Expectation-Maximization* (EM) algorithm [Blume 2002]. This algorithm consists of two steps, the *expectation* step where for each data $\mathbf{Y}_{\mathrm{LE}_n}$ in the set, with $n = 1, ...N$, a responsibility score denoted by $\gamma_k(\mathbf{Y}_{\mathrm{LE}_n})$ is assigned to each distribution $k, k = 1, ...K$ in the mixture model. This score indicates how much the multivariate Gaussian k is "responsible" for generating the data $\mathbf{Y}_{\mathrm{LE}_n}$. It is computed as the likelihood of the data sample $\mathbf{Y}_{\mathrm{LE}_n}$ to the k-th Gaussian distribution divided by the sum of the likelihoods of $\mathbf{Y}_{\mathrm{LE}_n}$ to all the K distributions in the mixture model. At the iteration i of the EM algorithm it holds:

$$\gamma_k^i(\mathbf{Y}_{\mathrm{LE}_n}) = \frac{\widehat{\omega}_k^{(i-1)} \hat{p}(\mathbf{Y}_{\mathrm{LE}_n}|\widehat{\boldsymbol{\mu}}_k^{(i-1)}, \widehat{\boldsymbol{\Sigma}}_k^{(i-1)})}{\sum_{j=1}^{K} \widehat{\omega}_j^{(i-1)} \hat{p}(\mathbf{Y}_{\mathrm{LE}_n}|\widehat{\boldsymbol{\mu}}_j^{(i-1)}, \widehat{\boldsymbol{\Sigma}}_j^{(i-1)})}. \tag{D.2}$$

The *maximisation* step of the EM algorithm follows, where the values of the estimates are recomputed given the scores obtained in the previous step. Thus, the parameter estimates of the

component $k, k = 1, ...K$ of the multivariate Gaussian mixture model at iteration i are given by:

$$\widehat{\boldsymbol{\mu}}_k^i = \frac{1}{N_k} \sum_{n=1}^{N} \gamma_k^i(\mathbf{Y}_{\mathrm{LE}_n}) \mathbf{Y}_{\mathrm{LE}_n}, \text{ where } N_k = \sum_{n=1}^{N} \gamma_k^i(\mathbf{Y}_{\mathrm{LE}_n}) \tag{D.3}$$

$$\widehat{\boldsymbol{\Sigma}}_k^i = \frac{1}{N_k} \sum_{n=1}^{N} \gamma_k^i(\mathbf{Y}_{\mathrm{LE}_n})(\mathbf{Y}_{\mathrm{LE}_n} - \widehat{\boldsymbol{\mu}}_k^i)(\mathbf{Y}_{\mathrm{LE}_n} - \widehat{\boldsymbol{\mu}}_k^i)^\dagger \tag{D.4}$$

$$\widehat{\omega}_k^i = \frac{N_k}{N}. \tag{D.5}$$

These steps are reiterated until a given convergence criterion or a number of maximum iterations is attained. The steps of the EM method for the estimation of the parameters of a multivariate Gaussian mixture model are given in the Algorithm 5.

Algorithm 5 *Expectation-Maximisation* algorithm for a multivariate Gaussian mixture model

Input: $\mathbf{Y}_{\mathrm{LE}_1}, ...\mathbf{Y}_{\mathrm{LE}_N}, K, Niter$
Output: $\widehat{\boldsymbol{\mu}}_k, \widehat{\boldsymbol{\Sigma}}_k, \widehat{\omega}_k, k = 1, ...K$
Begin
for $k = 1 : K$ **do**
  Initialise ω_k with $\frac{1}{K}$
  Initialise $\boldsymbol{\mu}_k$ with a random data sample from the input set $\mathbf{Y}_{\mathrm{LE}_1}, ...\mathbf{Y}_{\mathrm{LE}_N}$
  Initialise $\boldsymbol{\Sigma}_k$ according to equation D.4, for $N_k = N$ and $\gamma_k(\mathbf{Y}_{\mathrm{LE}_n}) = 1$
end for
$iter = 1$
while ($iter <= Niter$) or (no convergence) **do**
  for $k = 1 : K$ **do**
    Estimate $\widehat{\boldsymbol{\mu}}_k$ according to equation D.3
    Estimate $\widehat{\boldsymbol{\Sigma}}_k$ according to equation D.4
    Estimate $\widehat{\omega}_k$ according to equation D.5
  end for
  $iter = iter + 1$
end while
End

Appendix E

LST Multiscale Pyramid Computation

The multiscale pyramidal approaches proposed in this work rely on building a pyramid of LST fields computed at s scales of analysis. The lowest level of the pyramid, L_1, is associated to the highest resolution, namely to high frequency information (fine structural details). The highest pyramid level, L_s, corresponds to the lowest resolution, namely to low frequency information (macro structural information).

An LST field per scale of analysis is computed. This is achieved by fixing a value for the standard deviation of the Gaussian derivative kernel σ_G and tuning the standard deviation of the Gaussian weighting kernel σ_T for each of the s scales, with $\sigma_{T_1} < \sigma_{T_2} ... < \sigma_{T_s}$. Then, when passing from a level L_i to a level $L_{i+1}, i = 1, ..., s-1$, a decimation by a factor of 2 is applied in each dimension of the rectangular grid defining the sites of the LE LST field. Thus, at every level transition the number of points on the rectangular grid is reduced by 4.

When it comes to the LST computation at the different scales of analysis, we consider the following principles. A value σ_{T_1} adapted to the size of the textural pattern is chosen for the highest resolution level, L_1. Then, when passing to a lower resolution pyramid level, a new convolution is applied with a Gaussian weighting kernel of standard deviation α. Thus, for each pyramid level transition, a weighting kernel of constant standard deviation is applied. Let $W_1(0, \sigma_{T_1})$ and $W_\alpha(0, \alpha)$ be two 0-mean Gaussian weighting kernels corresponding to the first level of the pyramid and to a pyramid level transition, respectively. Thus, the Gaussian weighting kernel corresponding to the second pyramid level is given by:

$$W_2 = W_1 * W_\alpha \tag{E.1}$$

where $*$ stands for the convolution operator. Then, the following relation holds:

$$\sigma_2^2 = \sigma_{T_1}^2 + \alpha^2. \tag{E.2}$$

with σ_2 being the standard deviation of W_2.

Generalizing the previous relation, the standard deviation of the Gaussian weighting kernel ap-

plied at a level $i, i = 1, ...s$ of the pyramid is given by:

$$\sigma_i = \sqrt{\sigma_{T_1}^2 + (i-1)\alpha^2}. \tag{E.3}$$

Hence, in practice, instead of applying successive convolutions, one single convolution is applied at each pyramid level with a weighting kernel of standard deviation computed according to equation (E.3).

The value of the pyramid level transition standard deviation α induces the degree of resolution loss when passing from one pyramid level to the next one. In this work, a value of 1 has been chosen for the α parameter so that, for each pyramid level transition the structure tensor at a point is computed by considering mainly the information at the 8 neighbouring pixels, following a similar principle as the Laplacian pyramid approach [Burt 1987].

Bibliography

[Abdelghafour 2018] F. Abdelghafour, R. Rosu, B. Keresztes, C. Germain and J.P. Da Costa. *Joint structure and colour based parametric classification of grapevine organs from proximal images through several critical phenological stages.* In Proceedings of the 14th International Conference on Precision Agriculture, Montreal, Canada, Jun 2018. (Cited on page 162.)

[Aja-Fernández 2009] S. Aja-Fernández, D. de Luis-García R.and Tao and X. Li. Tensors in image processing and computer vision. Springer Publishing Company, Incorporated, 1st édition, 2009. (Cited on page 44.)

[Akl 2014] A. Akl, C. Yaacoub, M. Donias, J.-P. Da Costa and C. Germain. *Structure tensor based synthesis of directional textures for virtual material design*. In Proc. of ICIP, pages 4867–4871, Oct 2014. (Cited on page 44.)

[Akl 2015] A. Akl, C. Yaacoub, M. Donias, J.-P. Da Costa and C. Germain. *Texture Synthesis Using the Structure Tensor*. IEEE Trans. Image Process., vol. 24, no. 11, pages 4082–4095, November 2015. (Cited on pages v, 1, 9, 40, 44 and 51.)

[Akl 2016] A. Akl. *Analyse / synthèse de champs de tenseurs de structure : application à la synthèse d'images et de volumes texturés*. PhD thesis, 2016. (Cited on pages 26, 124, 125 and 138.)

[Akl 2018] A. Akl, C. Yaacoub, M. Donias, J.-P. Da Costa and C. Germain. *Synthesis of arbitrary-shaped textures constrained by the structure tensor field*. Signal, Image and Video Processing, vol. 12, no. 1, pages 41–49, Jan 2018. (Cited on pages 124, 125 and 162.)

[Amadasun 1989] M. Amadasun and R. King. *Textural features corresponding to textural properties*. IEEE Transactions on Systems, Man, and Cybernetics, vol. 19, no. 5, pages 1264–1274, Sep 1989. (Cited on pages 37 and 38.)

[Aptoula 2014] E. Aptoula. *Remote Sensing Image Retrieval With Global Morphological Texture Descriptors*. IEEE Trans. Geosci. Remote Sens., vol. 52, no. 5, pages 3023–3034, May 2014. (Cited on pages 39, 94, 96, 102, 103, 119 and 120.)

[Arseneau 2006] S. Arseneau and J. R. Cooperstock. *An Improved Representation of Junctions Through Asymmetric Tensor Diffusion*. In Advances in Visual Computing, pages 363–372, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg. (Cited on page 43.)

[Arsigny 2005] V. Arsigny, P. Fillard, X. Pennec and N. Ayache. *Fast and Simple Computations on Tensors with Log-Euclidean Metrics*. Research Report RR-5584, INRIA, 2005. (Cited on pages vii, 6, 18, 19 and 21.)

[Arsigny 2006] V. Arsigny, P. Fillard, X. Pennec and N. Ayache. *Log-Euclidean metrics for fast and simple calculus on diffusion tensors*. Magn. Reson. Med., vol. 56, no. 2, pages 411–421, August 2006. (Cited on pages 6, 9, 18, 19 and 21.)

[Atkinson 1981] Colin Atkinson and Ann F. S. Mitchell. *Rao's Distance Measure*. Sankhyā: The Indian Journal of Statistics, Series A (1961-2002), vol. 43, no. 3, pages 345–365, 1981. (Cited on pages 6 and 10.)

[Basser 1994] P.J. Basser, J. Mattiello and D. LeBihan. *MR diffusion tensor spectroscopy and imaging*. Biophysical Journal, vol. 66, no. 1, pages 259 – 267, 1994. (Cited on page 9.)

[Beguet 2012] B. Beguet, Nesrine Chehata, S. Boukir and D. Guyon. *Retrieving forest structure variables from Very High Resolution satellite images using an automatic method*. ISPRS Ann. Photogramm., Remote Sens. Spatial Inform. Sci., vol. 1, no. 7, pages 1–6, September 2012. (Cited on page 90.)

[Besag 1974] J. Besag. *Spatial Interaction and the Statistical Analysis of Lattice Systems*. Journal of the Royal Statistical Society. Series B (Methodological), vol. 36, no. 2, pages pp. 192–236, 1974. (Cited on page 125.)

[Besag 1986] J. Besag. *On the statistical analysis of dirty pictures*. JOURNAL OF THE ROYAL STATISTICAL SOCIETY B, vol. 48, no. 3, pages 48–259, 1986. (Cited on pages 125 and 127.)

[Bigün 1987] J. Bigün and G.H. Granlund. *Optimal Orientation Detection of Linear Symmetry*. In Proc. IEEE First Int. Conf. on Computer Vision, pages 433–438, London, Great Britain, 1987. (Cited on pages v, 1, 43 and 44.)

[Bishop 2006] Christopher M. Bishop. Pattern recognition and machine learning (information science and statistics). Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006. (Cited on page 9.)

[Blume 2002] M. Blume. *Expectation Maximization: A Gentle Introduction*. Tech. Univ. of Munich Institute for Comp. Sci., 2002. (Cited on pages 23 and 175.)

[Bombrun 2008] L. Bombrun and J. M. Beaulieu. *Fisher Distribution for Texture Modeling of Polarimetric SAR Data*. IEEE Geoscience and Remote Sensing Letters, vol. 5, no. 3, pages 512–516, July 2008. (Cited on page 9.)

[Bombrun 2011a] L. Bombrun, S. N. Anfinsen and O. Harant. *A complete coverage of log-cumulant space in terms of distributions for Polarimetric SAR data*. In 5th International Workshop on Science and Applications of SAR Polarimetry and Polarimetric Interferometry (POLinSAR 2011), pages 1–8, Friscati, Italy, January 2011. (Cited on page 43.)

[Bombrun 2011b] L. Bombrun, N. E. Lasmar, Y. Berthoumieu and G. Verdoolaege. *Multivariate texture retrieval using the SIRV representation and the geodesic distance*. In 2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 865–868, May 2011. (Cited on pages xi, 43 and 103.)

[Boser 1992] B. E. Boser and et al. *A training algorithm for optimal margin classifiers*. In Proc. 5th Annu. Workshop Computational Learning Theory, volume 5, pages 144–152, Pittsburgh, Pennsylvania, USA, July 1992. (Cited on page 98.)

[Boubchir 2010] L. Boubchir, A. Nait-Ali and E. Petit. *Multivariate statistical modeling of images in sparse multiscale transforms domain*. In 2010 IEEE International Conference on Image Processing, pages 1877–1880, Sept 2010. (Cited on page 41.)

[Brodatz 1966] P. Brodatz. *A Photographic Album for Artists and Designers*. `http://www.ux.uis.no/~tranden/brodatz.html`, 1966. (Cited on page 37.)

[Burt 1987] P. J. Burt and E. H. Adelson. *Readings in Computer Vision: Issues, Problems, Principles, and Paradigms*. chapter The Laplacian Pyramid As a Compact Image Code, pages 671–679. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1987. (Cited on page 178.)

[Castellano 2004] G. Castellano, L. Bonilha, L.M. Li and F. Cendes. *Texture analysis of medical images*. Clinical Radiology, vol. 59, no. 12, pages 1061 – 1069, 2004. (Cited on page 39.)

[Champion 2014] I. Champion, C. Germain, J.P. da Costa, A. Alborini and P. Dubois-Fernandez. *Retrieval of Forest Stand Age From SAR Image Texture for Varying Distance and Orientation Values of the Gray Level Co-Occurrence Matrix*. IEEE Geosci. Remote Sens. Lett., vol. 11, no. 1, pages 5–9, January 2014. (Cited on page 90.)

[Chang 2011] Chih-Chung Chang and Chih-Jen Lin. *LIBSVM: A library for support vector machines*. ACM Transactions on Intelligent Systems and Technology, vol. 2, pages 27:1–27:27, 2011. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`. (Cited on page 98.)

[Chanussot 2005] J. Chanussot, P. Bas and L. Bombrun. *Airborne remote sensing of vineyards for the detection of dead vine trees*. In Proc. IEEE Int. Geosci. Remote Sens. Symp., volume 5, pages 3090–3093, Seoul, Korea, July 2005. (Cited on page 39.)

[Choe 2012] B.-H. Choe, D. Kim, J.-H. Hwang, Y. Oh and W. Moon. *Detection of oyster habitat in tidal flats using multi-frequency polarimetric SAR data*. vol. 97, page 28–37, jan 2012. (Cited on page 89.)

[Cohen 1960] J. Cohen. *A Coefficient of Agreement for Nominal Scales*. Educ. Psychol. Meas., vol. 20, no. 1, pages 37–46, 1960. (Cited on page 100.)

[Congalton 1991] R. G. Congalton. *A review of assessing the accuracy of classifications of remotely sensed data*. Remote Sens. Environ., vol. 37, no. 1, pages 35 – 46, July 1991. (Cited on page 99.)

[Conners 1980] R. W. Conners and C. A. Harlow. *A Theoretical Comparison of Texture Algorithms*. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. PAMI-2, no. 3, pages 204–222, May 1980. (Cited on pages 36 and 40.)

[Conners 1984] R. W. Conners, M. M. Trivedi and C. A. Harlow. *Segmentation of a high-resolution urban scene using texture operators*. Computer Vision, Graphics, and Image Processing, vol. 25, no. 3, pages 273 – 310, 1984. (Cited on page 41.)

[Cross 1983] G. R. Cross and A. K. Jain. *Markov Random Field Texture Models*. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. PAMI-5, no. 1, pages 25–39, Jan 1983. (Cited on page 40.)

[Curran 1988] P. J. Curran. *The semivariogram in remote sensing: An introduction*. Remote Sensing of Environment, vol. 24, no. 3, pages 493 – 507, 1988. (Cited on page 40.)

[Da Costa 2007] J.-P. Da Costa, F. Michelet, C. Germain, O. Lavialle and G. Grenier. *Delineation of vine parcels by segmentation of high resolution remote sensed images*. Precision Agric., vol. 8, no. 1, pages 95–110, 2007. (Cited on page 39.)

[Da Costa 2012] J.-P. Da Costa, F. Galland, A. Roueff and C. Germain. *Unsupervised segmentation based on Von Mises circular distributions for orientation estimation in textured images*. J. Electron. Imaging, vol. 21, no. 2, pages 1–7, April 2012. (Cited on pages v, 1 and 51.)

[Da Costa 2015] J.P. Da Costa, P. Weisbecker, B. Farbos, J.-M. Leyssale, G.L. Vignoles and C. Germain. *Investigating carbon materials nanostructure using image orientation statistics*. Carbon, vol. 84, pages 160 – 173, 2015. (Cited on pages 92, 93 and 118.)

[Dav 1975] *A survey of edge detection techniques*. Computer Graphics and Image Processing, vol. 4, no. 3, pages 248 – 270, 1975. (Cited on page 44.)

[de Luis-García 2008] R. de Luis-García, R. Deriche and C. Alberola-López. *Texture and color segmentation based on the combined use of the structure tensor and the image components*. Signal Process., vol. 88, no. 4, pages 776–795, April 2008. (Cited on pages v, 1 and 9.)

[Do 2002] M.N. Do and M. Vetterli. *Wavelet-Based Texture Retrieval Using Generalized Gaussian Density and Kullback-Leibler Distance*. IEEE Trans. Image Process, vol. 11, no. 2, pages 146–158, February 2002. (Cited on pages 40, 41 and 95.)

[Donias 2007] M. Donias, C. David, Y. Berthoumieu, O. Lavialle, S. Guillon and N. Keskes. *New fault attribute based on robust directional scheme*. GEOPHYSICS, vol. 72, no. 4, pages P39–P46, 2007. (Cited on page 43.)

[Dryden 2009] I. L. Dryden, A. Koloydenko and D. Zhou. *Non-Euclidean Statistics for Covariance Matrices, with Applications to Diffusion Tensor Imaging.* Ann. Appl. Stat., vol. 3, no. 3, pages 1102–1123, September 2009. (Cited on pages vi, vii, 1, 7, 8 and 26.)

[Duncan 2000] J. S. Duncan and N. Ayache. *Medical Image Analysis: Progress over Two Decades and the Challenges Ahead.* IEEE Trans. Pattern Anal. Mach. Intell., vol. 22, no. 1, pages 85–106, 2000. (Cited on page 39.)

[Efros 1999] A. A. Efros and T. K. Leung. *Texture Synthesis by Non-Parametric Sampling*. In Proceedings of the International Conference on Computer Vision-Volume 2 - Volume 2, ICCV '99, pages 1033–, Washington, DC, USA, 1999. IEEE Computer Society. (Cited on pages 37 and 40.)

[Ennis 2006] D. B. Ennis and G. Kindlmann. *Orthogonal tensor invariants and the analysis of diffusion tensor magnetic resonance images*. Magnetic Resonance in Medicine, vol. 55, no. 1, pages 136–146, 2006. (Cited on pages ix, 26 and 45.)

[Fauvel 2007] Mathieu Fauvel. *Spectral and Spatial Methods for the Classification of Urban Remote Sensing Data*. Theses, Institut National Polytechnique de Grenoble - INPG ; Université d'Islande, November 2007. (Cited on page 99.)

[Förstner 1987] W. Förstner and E. Gülch. *A Fast Operator for Detection and Precise Location of Distinct Points, Corners and Centres of Circular Features*, 1987. (Cited on page 44.)

[Galerne 2011] B. Galerne, Y. Gousseau and J.-M. Morel. *Micro-Texture Synthesis by Phase Randomization*. Image Processing On Line, vol. 1, 2011. (Cited on pages 38 and 40.)

[Gatys 2015] L. A. Gatys, A. S. Ecker and M. Bethge. *Texture Synthesis Using Convolutional Neural Networks*. In Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1, NIPS'15, pages 262–270, Cambridge, MA, USA, 2015. MIT Press. (Cited on page 40.)

[Geman 1984] S. Geman and D. Geman. *Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images*. IEEE Trans. Pattern Anal. Mach. Intell., vol. 6, no. 6, pages 721–741, November 1984. (Cited on page 125.)

[Gentle 2009] J. E. Gentle. Computational statistics. Number 1. Springer-Verlag New York, 2009. (Cited on page 23.)

[Germain 2003] C. Germain, J.-P. Da Costa, O. Lavialle and P. Baylou. *Multiscale estimation of vector field anisotropy - application to texture characterization*. Signal Processing, vol. 83, no. 7, pages 1487–1503, 2003. (Cited on page 92.)

[Goodman 1963] N. R. Goodman. *Statistical Analysis Based on a Certain Multivariate Complex Gaussian Distribution (An Introduction)*. Ann. Math. Statist., vol. 34, no. 1, pages 152–177, March 1963. (Cited on page 8.)

[Gower 1975] J. C. Gower. *Generalized procrustes analysis*. Psychometrika, vol. 40, no. 1, pages 33–51, March 1975. (Cited on page 7.)

[Hajri 2016] H. Hajri, I. Ilea, S. Said, L. Bombrun and Y. Berthoumieu. *Riemannian Laplace Distribution on the Space of Symmetric Positive Definite Matrices*. Entropy, vol. 18, no. 3, 2016. (Cited on page 9.)

[Haralick 1973] R.M. Haralick, K. Shanmugam and Dinstein I. *Textural Features for Image Classification*. Systems Man and Cybernetics, vol. SMC-3, no. 6, pages 610–621, 1973. (Cited on pages xi, 36, 40, 41 and 102.)

[Haralick 1979] R. M. Haralick. *Statistical and structural approaches to texture*. Proceedings of the IEEE, vol. 67, no. 5, pages 786–804, May 1979. (Cited on pages xi, 36, 40, 83 and 102.)

[Harandi 2012] Mehrtash T. Harandi, Conrad Sanderson, Richard Hartley and Brian C. Lovell. *Sparse Coding and Dictionary Learning for Symmetric Positive Definite Matrices: A Kernel Approach*. In Proceedings of the 12th European Conference on Computer Vision - Volume Part II, ECCV'12, pages 216–229, Berlin, Heidelberg, 2012. Springer-Verlag. (Cited on page 9.)

[Harris 1988] C. Harris and M. Stephens. *A Combined Corner and Edge Detector*. In Proceedings of the 4th Alvey Vision Conference, pages 147–151, 1988. (Cited on pages v, 1 and 43.)

[Hastings 1970] W. K. Hastings. *Monte Carlo sampling methods using Markov chains and their applications*. Biometrika, vol. 57, no. 1, pages 97–109, 1970. (Cited on page 15.)

[Hayes 1974] K. C. Hayes, Shah A. N. and Rosenfeld A. *Texture coarseness: Further experiments*. In IEEE Trans. Syst. Man. Cybern., volume SMC-4, pages 467–472, Sept 1974. (Cited on page 37.)

[Hayes 2006] J. Hayes. *Wikipedia, the Free Encyclopedia*. https://commons.wikimedia.org/wiki/File:Texture_spectrum.jpg, 2006. Online; Accessed: 2018-01-28. (Cited on page 37.)

[Hershey 2007] J. R. Hershey and P. A. Olsen. *Approximating the Kullback Leibler Divergence Between Gaussian Mixture Models*. In IEEE International Conference on Acoustics, Speech and Signal Processing - ICASSP '07, volume 4, pages IV–317–IV–320, April 2007. (Cited on page 16.)

[Ilea 2015] I. Ilea, L. Bombrun, C. Germain, I. Champion, R. Terebes and M. Borda. *Statistical hypothesis test for maritime pine forest SAR images classification based on the geodesic distance*. In 2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), pages 3215–3218, July 2015. (Cited on page 102.)

[Ilea 2017] I. Ilea. *Robust classifcation methods on the space of covariance matrices. : application to texture and polarimetric synthetic aperture radar image classification*. PhD thesis, 2017. (Cited on pages 9, 12 and 41.)

[Jähne 1993] B. Jähne. Spatio-temporal image processing: Theory and scientific applications. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 1993. (Cited on page 43.)

[Jain 1990] A. K. Jain and F. Farrokhnia. *Unsupervised texture segmentation using Gabor filters*. In 1990 IEEE International Conference on Systems, Man, and Cybernetics Conference Proceedings, pages 14–19, Nov 1990. (Cited on page 42.)

[Jain 1992] A. K. Jain, S. K. Bhattacharjee and Y Chen. *On texture in document images*. In Proceedings 1992 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pages 677–680, Jun 1992. (Cited on page 40.)

[James 1973] A.T. James. *The Variance Information Manifold and the Functions on It*. 12 1973. (Cited on pages vii and 11.)

[Jayasumana 2013] S. Jayasumana, R. Hartley, M. Salzmann, Hongdong Li and M. Harandi. *Kernel Methods on the Riemannian Manifold of Symmetric Positive Definite Matrices*. In Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on, pages 73–80, June 2013. (Cited on pages vii, 6, 7 and 8.)

[Julesz 1962] B. Julesz. *Visual Pattern Discrimination*. IRE Transactions on Information Theory, vol. 8, no. 2, pages 84–92, February 1962. (Cited on pages 36 and 40.)

[Julesz 1978] B. Julesz, E. N. Gilbert and J. D. Victor. *Visual discrimination of textures with identical third-order statistics*. Biological Cybernetics, vol. 31, no. 3, pages 137–140, Sep 1978. (Cited on page 36.)

[Julesz 1983] B. Julesz and J.R. Bergen. *Textons, The Fundamentals Elements in Preattentive Vision and Perception of Textures*. Bell Syst. Tech. J., vol. 62, no. 6, pages 1619–1645, July 1983. (Cited on pages 36 and 40.)

[Kanevski 2009] M. Kanevski, A. Pozdnoukhov and V. Timonin. Machine learning for spatial environmental data: Theory, applications, and software. Engineering sciences. EFPL Press, 2009. (Cited on page 74.)

[Kass 1987] M. Kass and A. Witkin. *Analysing Oriented Patterns*. CVGIP:, vol. 37, pages 362–385, 1987. (Cited on page 44.)

[Kaufmann 2015] L. Kaufmann. *Classification d'images de télédétection par le tenseur de structure - application à la cartographie de parcelles forestières et ostréicoles*. Master 1 internship report, IMS Laboratory Bordeaux, 2015. (Cited on page 109.)

[Kayitakire 2006] F. Kayitakire, C. Hamel and P. Defourny. *Retrieving forest structure variables based on image texture analysis and IKONOS-2 imagery*. Remote Sens. Environ., vol. 102, no. 3–4, pages 390 – 401, June 2006. (Cited on page 90.)

[Kirkpatrick 1983] S. Kirkpatrick, C. D. Gelatt and M. P. Vecchi. *Optimization by Simulated Annealing*. SCIENCE, vol. 220, no. 4598, pages 671–680, 1983. (Cited on page 128.)

[Knutsson 1989] H. Knutsson. *Representing local structure using tensors*. In Proc. 6th Scand. Conf. Image Analysis, pages 244–251, Oulu, Finland, June 1989. (Cited on pages v, 1, 43, 44 and 46.)

[Kullback 1951] S. Kullback and R. A. Leibler. *On Information and Sufficiency*. The Annals of Mathematical Statistics, vol. 22, no. 1, pages 79–86, March 1951. (Cited on page 15.)

[Kwitt 2009] R. Kwitt and A. Uhl. *A Joint Model of Complex Wavelet Coefficients for Texture Retrieval*. In 16th IEEE Int. Conf. on Image Process., pages 1857–1860, Cairo, Egypt, November 2009. (Cited on pages 16 and 41.)

[Lasmar 2014] N. E. Lasmar and Y. Berthoumieu. *Gaussian Copula Multivariate Modeling for Texture Image Retrieval Using Wavelet Transforms*. IEEE Transactions on Image Processing, vol. 23, no. 5, pages 2246–2261, May 2014. (Cited on pages 43, 103 and 110.)

[Le Bris 2016] A. Le Bris, P. Rosa, A. Lerouxel, B. Cognie, P. Gernez, P. Launeau, M. Robin and B. Laurent. *Hyperspectral Remote Sensing Of Wild Oyster Reefs*. vol. 172, pages 1–12, jan 2016. (Cited on page 89.)

[Ledoit 2004] O. Ledoit and M. Wolf. *A well-conditioned estimator for large-dimensional covariance matrices*. J. Multivar. Anal., vol. 88, no. 2, pages 365–411, February 2004. (Cited on page 53.)

[Lee 1994] J. S. Lee, D. L. Schuler, R. H. Lang and K. J. Ranson. *K-distribution for multi-look processed polarimetric SAR imagery*. In Geoscience and Remote Sensing Symposium, 1994. IGARSS '94. Surface and Atmospheric Remote Sensing: Technologies, Data Analysis and Interpretation., International, volume 4, pages 2179–2181, Aug 1994. (Cited on page 8.)

[Lee 1997] J.M. Lee. Riemannian manifolds: An introduction to curvature. Graduate Texts in Mathematics. Springer New York, 1997. (Cited on page 6.)

[Lenglet 2006] C. Lenglet, M. Rousson, R. Deriche and O. Faugeras. *Statistics on the Manifold of Multivariate Normal Distributions: Theory and Application to Diffusion Tensor MRI Processing*. J. Math. Imaging Vis., vol. 25, no. 3, pages 423–444, August 2006. (Cited on pages vi, 1, 7, 9, 12 and 13.)

[Li 2001] S. Z. Li. Markov random field modeling in image analysis. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2001. (Cited on page 125.)

[Lyvers 1988] E. P. Lyvers and O. R. Mitchell. *Precision edge contrast and orientation estimation*. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 10, no. 6, pages 927–937, Nov 1988. (Cited on page 44.)

[Maillard 2003] P. Maillard. *Comparing Texture Analysis Methods through Classification*. Photogramm. Eng. Remote Sens., vol. 69, no. 4, pages 357–367, April 2003. (Cited on pages 41, 83 and 102.)

[Mallat 1989] S. G. Mallat. *A theory for multiresolution signal decomposition: the wavelet representation*. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 11, no. 7, pages 674–693, Jul 1989. (Cited on pages 40 and 42.)

[Manjunath 1996] B. S. Manjunath and W. Y. Ma. *Texture Features for Browsing and Retrieval of Image Data*. IEEE Trans. Pattern Anal. Mach. Intell., vol. 18, no. 8, pages 837–842, 1996. (Cited on page 42.)

[Manjunath 2001] B. S. Manjunath, J. R. Ohm, V. V. Vasudevan and A. Yamada. *Color and Texture Descriptors*. IEEE Trans. Cir. and Sys. for Video Technol., vol. 11, no. 6, pages 703–715, June 2001. (Cited on pages 96 and 97.)

[Marĉelja 1980] S. Marĉelja. *Mathematical description of the responses of simple cortical cells∗*. J. Opt. Soc. Am., vol. 70, no. 11, pages 1297–1300, Nov 1980. (Cited on page 42.)

[Mardia 2000] K. V. Mardia and Jupp P. E. Directional statistics. John Wiley & Sons, Ltd., England, 2000. (Cited on page v.)

[Materka 1998] A. Materka and M. Strzelecki. *Texture Analysis Methods - A Review*. 01 1998. (Cited on page 40.)

[Mathiassen 2002] J. R. Mathiassen, A. Skavhaug and K. Bø. *Texture Similarity Measure Using Kullback-Leibler Divergence between Gamma Distributions*. In Computer Vision — ECCV 2002, pages 133–147, Berlin, Heidelberg, 2002. Springer Berlin Heidelberg. (Cited on pages 40 and 41.)

[Metropolis 1953] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, Teller A. H. and E. Teller. *Equation of State Calculations by Fast Computing Machines*. The Journal of Chemical Physics, vol. 21, no. 6, pages 1087–1092, 1953. (Cited on pages 15 and 127.)

[MIT 1995] MIT. *VisTex Database of Color Textures*. http://vismod.media.mit.edu/vismod/imagery/VisionTexture/vistex.html, 1995. (Cited on page 37.)

[Mountrakis 2011] G. Mountrakis, J. Im and Ogole C. *Support vector machines in remote sensing: A review*. ISPRS Journal of Photogrammetry and Remote Sensing, vol. 66, no. 3, pages 247 – 259, 2011. (Cited on page 98.)

[Müller 2001] H. Müller, W. Müller, D. McG. Squire, S. Marchand-Maillet and T. Pun. *Performance evaluation in content-based image retrieval: overview and proposals*. Pattern Recogn. Lett., vol. 22, no. 5, pages 593 – 601, April 2001. (Cited on page 95.)

[Noutatiem Guiafaing 2016] Y. Noutatiem Guiafaing. *Modélisation spatiale et multi- échelle d'un champ de tenseur de structure - application en indexation d'images texturées*. Master 1 internship report, IMS Laboratory Bordeaux, 2016. (Cited on pages 82, 109 and 118.)

[Ojala 1994] T. Ojala, M. Pietikainen and D. Harwood. *Performance evaluation of texture measures with classification based on Kullback discrimination of distributions*. In Proceedings of 12th International Conference on Pattern Recognition, volume 1, pages 582–585, Oct 1994. (Cited on pages 40, 41 and 102.)

[Ojala 2002a] T. Ojala, T. Mäenpää, M. Pietikäinen, J. Viertola, J. Kyllönen and S. Huovinen. *Outex - New framework for empirical evaluation of texture analysis algorithms.* 2002. Proc. 16th International Conference on Pattern Recognition, Quebec, Canada, 1:701 - 706. (Cited on page 37.)

[Ojala 2002b] T. Ojala, M. Pietikäinen and T. Mäenpää. *Multiresolution Gray-Scale and Rotation Invariant Texture Classification with Local Binary Patterns.* IEEE Trans. Pattern Anal. Mach. Intell., vol. 24, no. 7, pages 971–987, July 2002. (Cited on pages xi, 42, 102 and 103.)

[Oxf 1992] *Oxford English Living Dictionaries*. https://en.oxforddictionaries.com/, 1992. Online; Accessed: 2018-01-26. (Cited on page 36.)

[Özkan 2014] S. Özkan, T. Ateş, E. Tola, M. Soysal and E. Esen. *Performance Analysis of State-of-the-Art Representation Methods for Geographical Image Retrieval and Categorization*.

IEEE Geosci. Remote Sens. Lett., vol. 11, no. 11, pages 1996–2000, November 2014. (Cited on pages 94, 96, 119 and 121.)

[Paget 1998] R. Paget and I. D. Longstaff. *Texture synthesis via a noncausal nonparametric multiscale Markov random field*. IEEE transactions on image processing : a publication of the IEEE Signal Processing Society, vol. 7, no. 6, pages 925–931, 1998. (Cited on page 157.)

[Papadakis 1999] N. G. Papadakis, D. Xing, G. C. Houston and J. M. Smith. *A study of rotationally invariant and symmetric indices of diffusion anisotropy*. Magnetic resonance ..., 1999. (Cited on page 45.)

[Pennec 2006a] X. Pennec. *Intrinsic Statistics on Riemannian Manifolds: Basic Tools for Geometric Measurements*. J. Math. Imaging Vis., vol. 25, no. 1, pages 127–154, 2006. (Cited on pages 6, 9 and 12.)

[Pennec 2006b] X. Pennec, P. Fillard and N. Ayache. *A Riemannian Framework for Tensor Computing*. International Journal of Computer Vision, vol. 66, no. 1, pages 41–66, 2006. (Cited on pages 9 and 11.)

[Perona 1990] P. Perona and J. Malik. *Scale-space and edge detection using anisotropic diffusion*. IEEE Trans. Pattern Anal. Mach. Intell., vol. 12, no. 7, pages 629–639, Jul 1990. (Cited on page 44.)

[Petrou 2006] M. Petrou and P. G. Sevilla. Image processing: Dealing with texture. John Wiley and Sons, 2006. (Cited on page 35.)

[Peyré 2010] G. Peyré. *Texture Synthesis with Grouplets*. IEEE Trans. Pattern Anal. Mach. Intell., vol. 32, no. 4, pages 733–746, 2010. (Cited on page 44.)

[Pham 2015a] M.-T. Pham, G. Mercier and J. Michel. *Covariance-based Texture Description from Weighted Coherency Matrix and Gradient Tensor for Polarimetric SAR Image Classification*. In Proc. IEEE Int. Geosci. Remote Sens. Symp., pages 2469–2472, Milan, Italy, July 2015. (Cited on page 44.)

[Pham 2015b] M.-T. Pham, G. Mercier and J. Michel. *Pointwise Approach on Covariance Matrix of Oriented Gradients for Very High Resolution Image Texture Segmentation*. In Proc. IEEE Int. Geosci. Remote Sens. Symp., pages 1008–1011, Milan, Italy, July 2015. (Cited on pages v, 1 and 44.)

[Pham 2016a] M. T. Pham, G. Mercier and J. Michel. *PW-COG: An Effective Texture Descriptor for VHR Satellite Imagery Using a Pointwise Approach on Covariance Matrix of Oriented Gradients*. IEEE Transactions on Geoscience and Remote Sensing, vol. 54, no. 6, pages 3345–3359, June 2016. (Cited on pages 9 and 102.)

[Pham 2016b] M. T. Pham, G. Mercier, O. Regniers, L. Bombrun and J. Michel. *Texture retrieval from very high resolution remote sensing images using local extrema-based descriptors*. In 2016 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), pages 1839–1842, jul 2016. (Cited on pages 89 and 102.)

[Rao 1991] A.R. Rao and B.G. Schunck. *Computing Oriented Texture Fields*. CVGIP: Graphical Models and Image Processing, vol. 53, no. 2, pages 157–185, 1991. (Cited on page 44.)

[Rao 1993] A.R. Rao and G.L. Lohse. *Identifying High Level Features of Texture Perception*. CVGIP: Graph. Models Image Process., vol. 55, no. 3, pages 218–233, May 1993. (Cited on page 38.)

[Regniers 2013] O. Regniers, J.-P. Da Costa, G. Grenier, C. Germain and L. Bombrun. *Texture based image retrieval and classification of very high resolution maritime pine forest images*. In Proc. IEEE Int. Geosci. Remote Sens. Symp., pages 4038–4041, Melbourne, Australia, July 2013. (Cited on pages 96 and 102.)

[Regniers 2014a] O. Regniers. *Méthodes d'analyse de texture pour la cartographie d'occupations du sol par télédétection THR - Application à la forêt, la vigne et les parcs ostréicoles*. PhD thesis, Université de Bordeaux, 2014. (Cited on pages 43, 82, 88, 90, 91, 92, 96, 98, 102, 103, 109, 110, 112, 113, 117 and 165.)

[Regniers 2014b] O. Regniers, L. Bombrun, D. Guyon, J.-C. Samalens, C. Tinel, G. Grenier and C. Germain. *Wavelet based texture modeling for the classification of very high resolution maritime pine forest images*. In Proc. IEEE Int. Geosci. Remote Sens. Symp., pages 2027–2030, Quebec, Canada, July 2014. (Cited on pages xi, 40, 90 and 112.)

[Regniers 2014c] O. Regniers, L. Bombrun, V. Lafon, A. Dehouck, C. Tinel and C. Germain. *Wavelet based texture modeling for panchromatic very high resolution image classification: Application to oyster racks detection*. In Proc. IEEE Int. Geosci. Remote Sens. Symp., pages 5148–5151, Quebec, Canada, July 2014. (Cited on pages xi, 40, 43, 89, 103 and 107.)

[Regniers 2015a] O. Regniers, L. Bombrun, D. Guyon, J.-C. Samalens and C. Germain. *Wavelet-Based Texture Features for the Classification of Age Classes in a Maritime Pine Forest*. IEEE Geosci. Remote Sens. Lett., vol. 12, no. 3, pages 621–625, March 2015. (Cited on pages 41, 100, 101, 102, 112, 113 and 117.)

[Regniers 2015b] O Regniers, L. Bombrun, I. Ilea, V. Lafon and C. Germain. *Classification of oyster habitats by combining wavelet-based texture features and polarimetric SAR descriptors*. In 2015 IEEE International Geoscience and Remote Sensing Symposium, IGARSS 2015, Milan, Italy, July 26-31, 2015, pages 3890–3893. IEEE, 2015. (Cited on page 39.)

[Regniers 2016] O. Regniers, L. Bombrun, V. Lafon and C. Germain. *Supervised Classification of Very High Resolution Optical Images Using Wavelet-Based Textural Features*. IEEE Trans. Geosci. Remote Sens., vol. 54, no. 6, pages 3722–3735, June 2016. (Cited on pages 39, 98 and 102.)

[Rosu 2013] R. G. Rosu and C. Rusu. *A sparsogram implementation for wildlife intruder detection*. In International Symposium on Signals, Circuits and Systems ISSCS2013, pages 1–4, Iasi, Romania, Jul 2013. (Not cited.)

[Rosu 2015a] R. Rosu, L. Bombrun, S. Said, O. Regniers, M. Donias and J.P. Da Costa. *Modèles probabilistes Riemanniens de tenseurs de structure pour le traitement d'images texturées*. In XXVème Colloque GRETSI - Group d'Etude du Traitement du Signal et des Images, Lyon, France, Sept 2015. (Cited on pages 9, 10, 13, 14, 108 and 173.)

[Rosu 2015b] R. G. Rosu, J. F. Giovannelli, A. Giremus and C. Vacar. *Potts model parameter estimation in Bayesian segmentation of piecewise constant images*. In 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 4080–4084, Brisbane, Queensland, Apr 2015. (Not cited.)

[Rosu 2016] R. G. Rosu, J. P. Da Costa and M. Donias. *Structure tensor Log-Euclidean statistical models for texture analysis*. In IEEE International Conference on Image Processing (ICIP), pages 3553–3557, Phoenix, Arizona, Sept 2016. (Cited on pages 9, 108 and 118.)

[Rosu 2017] R. Rosu, M. Donias, L. Bombrun, S. Said, O. Regniers and J. P. Da Costa. *Structure Tensor Riemannian Statistical Models for CBIR and Classification of Remote Sensing Images*. IEEE Transactions on Geoscience and Remote Sensing, vol. 55, no. 1, pages 248–260, Jan 2017. (Cited on pages 9 and 10.)

[Rousson 2003] M. Rousson, T. Brox and R. Deriche. *Active unsupervised texture segmentation on a diffusion based feature space*. In Proc. of CVPR, volume 2, pages 699–704, 2003. (Cited on pages v, 1 and 44.)

[Rouzaud 2002] J.-N. Rouzaud and C. Clinard. *Quantitative high-resolution transmission electron microscopy: a promising tool for carbon materials characterization*. Fuel Processing Technology, vol. 77, pages 229 – 235, 2002. (Cited on page 92.)

[Ruiz 2004] L. A. Ruiz, A. Fdez-Sarria and J. A. Recio. *Texture feature extraction for classification of remote sensing data using wavelet decomposition: a comparative study*. In Int. Archchives Photogramm. Remote Sens. Spatial Inform. Sci., pages 1109–1115, Istanbul, Turkey, July 2004. (Cited on page 39.)

[Said 2015] S. Said, L. Bombrun and Y. Berthoumieu. *Texture Classification Using Rao's Distance on the Space of Covariance Matrices*. In Geometric Sci. of Inform., pages 371–378, Paris, France, October 2015. (Cited on pages 10 and 14.)

[Said 2017] S. Said, L. Bombrun, Y. Berthoumieu and J. H. Manton. *Riemannian Gaussian Distributions on the Space of Symmetric Positive Definite Matrices*. IEEE Transactions on Information Theory, vol. 63, no. 4, pages 2153–2170, April 2017. (Cited on pages vii, 6, 9, 10, 11, 12, 14, 173 and 174.)

[Said 2018] S. Said, H. Hajri, L. Bombrun and B. C. Vemuri. *Gaussian Distributions on Riemannian Symmetric Spaces: Statistical Learning With Structured Covariance Matrices*. IEEE Trans. Information Theory, vol. 64, no. 2, pages 752–772, 2018. (Cited on pages 10 and 11.)

[Scholkopf 2001] Bernhard Scholkopf and Alexander J. Smola. Learning with kernels: Support vector machines, regularization, optimization, and beyond. MIT Press, Cambridge, MA, USA, 2001. (Cited on page 99.)

[Schwarz 1978] G. Schwarz. *Estimating the dimension of a model*. The Annals of Statistics, vol. 6, pages 461–464, 1978. (Cited on page 66.)

[Shim 2000] H.-S. Shim, R. H. Hurt and N. Y.C. Yang. *A methodology for analysis of 002 lattice fringe images and its application to combustion-derived carbons*. Carbon, vol. 38, no. 1, pages 29 – 45, 2000. (Cited on page 92.)

[Silverman 1986] B. W. Silverman. Density estimation for statistics and data analysis. Chapman & Hall, London, 1986. (Cited on pages 157 and 162.)

[Sklansky 1977] J. Sklansky. Image segmentation and feature extraction. Pattern recognition research: Technical report. School of Engineering, University of California, 1977. (Cited on page 37.)

[Srivastava 2002] A. Srivastava, X. Liu and U. Grenander. *Universal Analytical Forms for Modeling Image Probabilities.* vol. 24, pages 1200–1214, 09 2002. (Cited on page 41.)

[Stillwell 2008] J. Stillwell. Naive lie theory. Undergraduate Texts in Mathematics. Springer New York, 2008. (Cited on page 19.)

[Stitou 2009] Y. Stitou, N. Lasmar and Y. Berthoumieu. *Copulas based multivariate gamma modeling for texture classification*. In 2009 IEEE International Conference on Acoustics, Speech and Signal Processing, pages 1045–1048, April 2009. (Cited on page 43.)

[Tamura 1978] H. Tamura, S. Mori and T. Yamawaki. *Textural Features Corresponding to Visual Perception*. IEEE Transactions on Systems, Man, and Cybernetics, vol. 8, no. 6, pages 460–473, June 1978. (Cited on pages 37 and 38.)

[Tarantola 2004] A. Tarantola. Inverse problem theory and methods for model parameter estimation. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2004. (Cited on page 74.)

[Terras 1988] A. Terras. Harmonic analysis on symmetric spaces and applications. Number v. 1. Springer-Verlag, 1988. (Cited on page 11.)

[Tomczak 2007] L. Tomczak, V. Mosorov, D. Sankowski and J. Nowakowski. *Image Defect Detection Methods for Visual Inspection Systems*. In 2007 9th International Conference - The Experience of Designing and Applications of CAD Systems in Microelectronics, pages 454–456, Feb 2007. (Cited on page 39.)

[Toth 2013] P. Toth, A. B. Palotas, E. G. Eddings, R. T. Whitaker and J. S. Lighty. *A novel framework for the quantitative analysis of high resolution transmission electron micrographs of soot II. Robust multiscale nanostructure quantification*. Combustion and Flame, vol. 160, no. 5, pages 920–932, 5 2013. (Cited on page 92.)

[Toujas 2010] V. Toujas, M. Donias and Y. Berthoumieu. *Structure Tensor Field Regularization Based on Geometric Features*. In 18th Eur. Signal Process. Conf., pages 1330–1334, Aalborg, Denmark, August 2010. (Cited on pages 44 and 51.)

[Tuceryan 1993] M. Tuceryan and A. K. Jain. *Handbook of Pattern Recognition &Amp; Computer Vision*. chapter Texture Analysis, pages 235–276. World Scientific Publishing Co., Inc., River Edge, NJ, USA, 1993. (Cited on page 40.)

[Turner 1986] M. R. Turner. *Texture discrimination by Gabor functions*. Biological Cybernetics, vol. 55, no. 2, pages 71–82, Nov 1986. (Cited on page 40.)

[Tuzel 2006] O. Tuzel, F. Porikli and P. Meer. Region covariance: A fast descriptor for detection and classification, pages 589–600. Springer Berlin Heidelberg, Berlin, Heidelberg, 2006. (Cited on page 9.)

[Tuzel 2008] O. Tuzel, F. Porikli and P. Meer. *Pedestrian Detection via Classification on Riemannian Manifolds*. IEEE Trans. Pattern Anal. Mach. Intell., vol. 30, no. 10, pages 1713–1727, October 2008. (Cited on page 9.)

[Unser 1984] M. Unser. *Description statistique de textures application à l'inspection automatique*. page 212, 1984. (Cited on page 37.)

[Urs 2012] R. Urs, J.-P. Da Costa, G. V. Leyssale and C. Germain. *Non-Parametric Synthesis of Laminar Volumetric Textures*. Proc. Brit. Mach. Vis. Conf., pages 54.1–54.11, 2012. (Cited on page 130.)

[Urs 2013] R. D. Urs. *Non-parametric synthesis of volumetric textures from a 2D sample*. Theses, March 2013. (Cited on pages 40, 138 and 157.)

[Vapnik 1979] V. N. Vapnik. Estimation of dependences based on empirical data [in russian]. Nauka, USSR, 1979. (Cited on page 98.)

[Verdoolaege 2011] G. Verdoolaege and P. Scheunders. *Geodesics on the Manifold of Multivariate Generalized Gaussian Distributions with an Application to Multicomponent Texture Discrimination*. International Journal of Computer Vision, vol. 95, no. 3, pages 265–286, May 2011. (Cited on page 41.)

[Wei 2000] L.-Y. Wei and M. Levoy. *Fast Texture Synthesis Using Tree-structured Vector Quantization*. In Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH '00, pages 479–488, New York, NY, USA, 2000. ACM Press/Addison-Wesley Publishing Co. (Cited on pages 40, 125, 129, 130 and 138.)

[Wei 2009] L.-Y. Wei, S. Lefebvre, V. Kwatra and G. Turk. *State of the Art in Example-based Texture Synthesis*. In Eurographics 2009, State of the Art Report, EG-STAR. Eurographics Association, 2009. (Cited on page 36.)

[Weickert 2014] J. Weickert and H. Hagen. Visualization and processing of tensor fields. Springer Publishing Company, Incorporated, 2014. (Cited on pages v and 47.)

[Wishart 1928] J. Wishart. *The generalised product moment distribution in samples from a normal multivariate population*. Biometrika, vol. 20A, no. 1/2, pages 32–52, 1928. (Cited on page 8.)

[Won 2004] C. S. Won and R. M Gray. Stochastic image processing. Springer US, 1st édition, 2004. (Cited on page 127.)

[Yang 2010] L. Yang. *Riemannian median and its estimation*. LMS Journal of Computation and Mathematics, vol. 13, page 461–479, 2010. (Cited on page 9.)

[Yang 2013] Y. Yang and S. Newsam. *Geographic Image Retrieval Using Local Invariant Features*. IEEE Trans. Geosci. Remote Sens., vol. 51, no. 2, pages 818–832, February 2013. (Cited on pages 94, 96, 97, 98, 119 and 120.)

[Yao 1973] K. Yao. *A representation theorem and its applications to spherically-invariant random processes*. IEEE Transactions on Information Theory, vol. 19, no. 5, pages 600–608, September 1973. (Cited on page 43.)

[Zhou 2001] F. Zhou, J. F. Feng and Q. Y. Shi. *Texture feature based on local Fourier transform*. In Proceedings 2001 International Conference on Image Processing (Cat. No.01CH37205), volume 2, pages 610–613 vol.2, Oct 2001. (Cited on page 40.)

List of Publications

Publications Related to the Thesis

Journal Paper

R. Rosu, M. Donias, L. Bombrun, S. Said, O. Regniers and J. P. Da Costa. *Structure Tensor Riemannian Statistical Models for CBIR and Classification of Remote Sensing Images*. IEEE Transactions on Geoscience and Remote Sensing, vol. 55, no. 1, pages 248–260, Jan 2017

Conference Papers

1. R. G. Rosu, J. P. Da Costa and M. Donias. *Structure tensor Log-Euclidean statistical models for texture analysis*. In IEEE International Conference on Image Processing (ICIP), pages 3553–3557, Phoenix, Arizona, Sept 2016

2. R. Rosu, L. Bombrun, S. Said, O. Regniers, M. Donias and J.P. Da Costa. *Modèles probabilistes Riemanniens de tenseurs de structure pour le traitement d'images texturées*. In XXVème Colloque GRETSI - Group d'Etude du Traitement du Signal et des Images, Lyon, France, Sept 2015

3. F. Abdelghafour, R. Rosu, B. Keresztes, C. Germain and J.P. Da Costa. *Joint structure and colour based parametric classification of grapevine organs from proximal images through several critical phenological stages.* In Proceedings of the 14th International Conference on Precision Agriculture, Montreal, Canada, Jun 2018

Other Publications (Conference Papers)

1. R. G. Rosu, J. F. Giovannelli, A. Giremus and C. Vacar. *Potts model parameter estimation in Bayesian segmentation of piecewise constant images*. In 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 4080–4084, Brisbane, Queensland, Apr 2015

2. R. G. Rosu and C. Rusu. *A sparsogram implementation for wildlife intruder detection*. In International Symposium on Signals, Circuits and Systems ISSCS2013, pages 1–4, Iasi, Romania, Jul 2013

Parametric Approaches for Modelling Local Structure Tensor Fields with Applications to Texture Analysis

Abstract: This thesis proposes and evaluates parametric frameworks for modelling local structure tensor (LST) fields computed on textured images.

A texture's underlying geometry is described in terms of orientation and anisotropy, estimated in each pixel by the LST. Defined as symmetric non-negative definite matrices, LSTs cannot be handled using the classical tools of Euclidean geometry. In this work, two complete Riemannian statistical frameworks are investigated to address the representation of symmetric positive definite matrices. They rely on the affine-invariant (AI) and log-Euclidean (LE) metric spaces. For each framework, a Gaussian distribution and its corresponding mixture models are considered for statistical modelling. Solutions for parameter estimation are provided and parametric dissimilarity measures between statistical models are proposed as well.

The proposed statistical frameworks are first considered for characterising LST fields computed on textured images. Both AI and LE models are first employed to handle marginal LST distributions. Then, LE models are extended to describe joint LST distributions with the purpose of characterising both spatial and multiscale dependencies. The theoretical models' fit to empirical LST distributions is experimentally assessed for a texture set composed of a large diversity of patterns.

The descriptive potential of the proposed statistical models are then assessed in two applications.

A first application consists of texture recognition. It deals with very high resolution remote sensing images and carbonaceous material images issued from high resolution transmission electron microscopy technology. The LST statistical modelling based approaches for texture characterisation outperform, in most cases, the state of the art methods. Competitive texture classification performances are obtained when modelling marginal LST distributions on both AI and LE metric spaces. When modelling joint LST distributions, a slight gain in performance is obtained with respect to the case when marginal distributions are modelled. In addition, the LST based methods' intrinsic ability to address the rotation invariance prerequisite that arises in many classification tasks dealing with anisotropic textures is experimentally validated as well. In contrast, state of the art methods achieve a rather *pseudo* rotation invariance.

A second application concerns LST field synthesis. To this purpose, monoscale and multiscale pyramidal approaches relying on a Markovian hypothesis are developed. Experiments are carried out on toy LST field examples and on real texture LST fields. The successful synthesis results obtained when optimal parameter configurations are employed, are a proof of the real descriptive potential of the proposed statistical models. However, the experiments have also shown a high sensitivity to the parameters' choice, that may be due to statistical inference limitations in high dimensional spaces.

Keywords: *local structure tensor, non-Euclidean metrics, covariance matrix statistical models, texture analysis, texture classification, Markov Random Fields, synthesis.*

UMR 5218 IMS - *Laboratoire de l'Intégration du Matériau au Système* - Group Signal,

351 Cours de la Libération, 33405 Talence cedex, France.

https://www.ims-bordeaux.fr

Approches Paramétriques pour la Modélisation de Champs de Tenseurs de Structure Locaux et Applications en Analyse de Texture

Résumé: Cette thèse porte sur des canevas méthodologiques paramétriques pour la modélisation de champs de tenseurs de structure locaux (TSL) calculés sur des images texturées.

Estimé en chaque pixel, le tenseur de structure permet la caractérisation de la géométrie d'une image texturée à travers des mesures d'orientation et d'anisotropie locales. Matrices symétriques semi-définies positives, les tenseurs de structure ne peuvent pas être manipulés avec les outils classiques de la géométrie euclidienne. Deux canevas statistiques riemanniens, reposant respectivement sur les espaces métriques affine invariant (AI) et log-euclidien (LE), sont étudiés pour leur représentation. Dans chaque cas, un modèle de distribution gaussienne et de mélange associé sont considérés pour une analyse statistique. Des algorithmes d'estimation de leurs paramètres sont proposés ainsi qu'une mesure de dissimilarité.

Les modèles statistiques proposés sont tout d'abord considérés pour décrire des champs de TSL calculés sur des images texturées. Les modèles AI et LE sont utilisés pour décrire des distributions marginales de TSL tandis que les modèles LE sont étendus afin de décrire des distributions jointes de TSL et de caractériser des dépendances spatiales et multi-échelles. L'ajustement des modèles théoriques aux distributions empiriques de TSL est évalué de manière expérimentale sur un ensemble de textures composées d'un spectre assez large de motifs structuraux.

Les capacités descriptives des modèles statistiques proposés sont ensuite éprouvées à travers deux applications.

Une première application concerne la reconnaissance de texture sur des images de télédétection très haute résolution et sur des images de matériaux carbonés issues de la microscopie électronique à transmission haute résolution. Dans la plupart des cas, les performances des approches proposées sont supérieures à celles obtenues par les méthodes de l'état de l'art. Sur l'espace LE, les modèles joints pour la caractérisation des dépendances spatiales au sein d'un champ de TSL améliorent légèrement les résultats des modèles opérant uniquement sur les distributions marginales. La capacité intrinsèque des méthodes basées sur le tenseur de structure à prendre en considération l'invariance à la rotation, requise dans beaucoup d'applications portant sur des textures anisotropes, est également démontrée de manière expérimentale.

Une deuxième application concerne la synthèse de champs de TSL. A cet effet, des approches mono-échelle ainsi que des approches pyramidales multi-échelles respectant une hypothèse markovienne sont proposées. Les expériences sont effectuées à la fois sur des champs de TSL simulés et sur des champs de TSL calculés sur des textures réelles. Efficientes dans quelques configurations et démontrant d'un potentiel réel de description des modèles proposés, les expériences menées montrent également une grande sensibilité aux choix des paramètres qui peut s'expliquer par des instabilités d'estimation sur des espaces de grande dimension.

Mots clés : *tenseur de structure local, métriques non-Euclidiennes, modèles statistiques de matrices de covariance, analyse de texture, classification de texture, champ aléatoire de Markov, synthèse*

UMR 5218 IMS - *Laboratoire de l'Intégration du Matériau au Système* - Group Signal,

351 Cours de la Libération, 33405 Talence cedex, France.

https://www.ims-bordeaux.fr