# Light Field Image and Video Compression

Nader Bakir

# THESE DE DOCTORAT DE

L'INSTITUT NATIONAL DES SCIENCES APPLIQUEES RENNES

ECOLE DOCTORALE N° 601
*Mathématiques et Sciences et Technologies de l'Information et de la Communication*
Spécialité : *(Signal, Image et Vision)*

Par

**Nader Bakir**

# Light Field Image and Video Compression

**Thèse présentée et soutenue à Rennes, le 10 Juin 2020**
**Unité de recherche : IETR - UMR CNRS 6164**
Thèse N° : 20ISAR 07 / D20 - 07

## Composition du Jury :

**Président**
Bachar ELHASSAN Professeur à l'Université Libanaise

**Rapporteurs**
Frédéric DUFAUX Directeur de Recherche, CNRS, Paris
Bachar ELHASSAN Professeur à l'Université Libanaise

**Examinateur**
Marco CAGNAZZO Professeur, Telecom Paris

**Directeur de thèse**
Olivier DEFORGES Professeur, IETR/INSA, France

**Co-directeur de thèse**
Mohamad KHALIL Professeur à l'Université Libanaise

**Encadrant de thèse**
Khouloud SAMROUTH MCF, Université Libanaise

**Invité (Encadrant de thèse)**
Wassim HAMIDOUCHE MCF, INSA de Rennes

UNIVERSITE BRETAGNE LOIRE / MATHSTIC

INSA RENNES

Intitulé de la thèse :

# Light Field Image and Video Compression

**Nader BAKIR**

En partenariat avec :

| | | | | |
|---|---|---|---|---|
| | | | | |

## Remerciements

# Contents

# Résumé en Francais

## Introduction

Les applications de vision par ordinateur telles que le refocusing, la segmentation et la classification deviennent l'un des services les plus avancés dans le domaine de traitement d'image. Dans telles applications nécessitent des informations sémantiques riches de la scène. La technologie 3D est largement utilisée dans les domaines du divertissement, de l'imagerie médicale et de l'éducation. Il existe différentes manières de représenter l'information 3D. L'une des plus répandues consiste à associer à une image classique dite de texture, une image de profondeur de champ. Cette représentation conjointe permet ainsi une bonne reconstruction 3D dès que les deux images sont bien corrélées, et plus particulièrement sur les zones de contours de l'image de profondeur. En comparaison avec des images 2D classiques, la connaissance de la profondeur de champs pour les images 3D apporte donc une information sémantique importante quant à la composition de la scène.

Figure 1: La technologie Light Field permet une reproduction de la réalité très fidèle en Réalité Virtuelle (VR)

Une autre technologie qui prend plus d'importance c'est la technologie LF. L'image

LF est une image non conventionnelle contenant des informations beaucoup plus que l'intensité sur les rayons lumineux qui interagissent avec la scène. Elle donne une description très riche d'une scène 3D permettant d'offrir une large bande de fonctionnalités. A titre d'exemple, elle permet la synthèse avancée de vues intermédiaires, ainsi que la re-focalisation de l'image après acquisition.

L'intérêt dans la technologie LF continue à croitre de manière très significative, notamment avec la pénétration croissante des dispositifs d'acquisition et d'affichage pour le contenu LF sur le marché du grand public.

En particulier, on utilise un ensemble dense de caméras et de matrices de microlentilles comme la caméra Plénoptique [NLB+05] (Figure 2), pour avoir la direction de chaque rayon venant de la scène vers le système d'acquisition LF.


(a) Lytro Illum (b) Raytrix

Figure 2: Exemples de caméras plénoptiques

Ceci peut être extrait et représenté par des coordonnées spatiales et angulaires. Cependant, un tel système d'imagerie présente de nombreux inconvénients, notamment la grande quantité de données produites et la complexité augmente pour la représentation de la scène. Ce qui pose donc de manière urgente la question de leur compression.

## Contributions

Dans cette thèse, nous proposons ainsi dans un premier temps un schéma de codage LF basé réseaux de neurones convolutionnels (CNN) qui inclut une optimisation débit-distorsion (RDO) suivi par un post-traitement. Le principe consiste à exploiter la corrélation entre les différentes vues LF et éviter le codage de toutes les vues. Les vues LF sont donc divisées en 3 ensembles, un premier ensemble qui est codé par un codeur 2D standard, un deuxième ensemble qui est approximé linéairement et un troisième ensemble qui sera synthétisé au niveau du décodeur soit par une approximation linéaire soit par synthèse avec Convolutional Neural Networks (CNN) selon la décision faite par le bloc RDO. Ensuite, nous intégrons le Dual Discriminator Generative Adversarial Nets (D2GAN) avec l'encodeur hiérarchique Versatile Video Coding (VVC). L'idée globale consiste à éviter de coder les vues de niveau hiérarchique supérieur et de les générer avec D2GAN au niveau du décodeur. Enfin, nous évaluons les deux schémas proposés subjectivement sur un ensemble d'images LF de plusieurs bases de données différentes. Les conditions des tests psychovisuels respectent les normes de l'Union Internationale

des Télécommunications (ITU). L'élément le plus remarquable est que les méthodes de codage basées sur la synthèse de vues peuvent atteindre des performances de codage élevées et démontrer leur efficacité en fournissant la meilleure qualité visuelle par rapport aux deux autres méthodes.

Ainsi, cette thèse est constituté de cinq chapitres et tente de développer des méthodes pour une compression efficace des images et vidéos basés LF:

Afin d'établir le contexte de cette thèse, le premier chapitre, nous faisons l'état de l'art des caractéristiques de l'image LF et des méthodes de compression existantes dans la littérature. L'image Light Field est une image non conventionnelle, contenant des informations beaucoup plus que l'intensité sur les rayons lumineux qui interagissent avec la scène. L'image LF donne une description très riche d'une scène 3D Permettant d'offrir une large bande de fonctionnalités. Notamment, elle permet la synthèse avancée de vues intermédiaires, ainsi que la re-focalisation de l'image après acquisition.

Il existe deux systèmes d'acquisitions d'images LF, le premier type est le système composé de multiples caméras conventionnelles bien alignées avec parallaxe horizontal ou bien avec parallaxe horizontal et vertical. Un tel système est appelé système super multi caméras et le deuxième type est le système LF Plenoptique. L'imagerie Plénoptique est limitée par certaines contraintes: taille très grande de l'image LF (50MB par scène), répétition des motifs, ceci rend le codage des images LF très coûteux en terme de calcul et de temps.

Le deuxième chapitre fournit une compréhension globale sur l'apprentissage en profond, les standards 2D de compression, les différentes méthodes existantes de codage d'images LF, enfin les métriques objectifs de la qualité et l'environnement du test subjectif. Nous nous concentrons sur l'apprentissage en profond qui a transformé la recherche en intelligence artificielle surtout pour la vision par ordinateur. puis nous introduisons les normes de codage vidéo High Efficiency Video Coding (HEVC) et VVC. Ensuite, nous analysons les différentes techniques de codage d'images LF existantes. Il existe plusieurs approches qui s'appliquent sur les diverses représentations du champ lumineux (e.g. image brute LF, sub apertures, épipolaire) A titre d'exemples, le codage basé Pseudo-séquence et le codage prédictif. Le troisième chapitre a proposé un nouveau schéma de codage d'image Light Field. Dans ce schéma, on considère l'image LF avec 8*8 vues subapertures. Les vues sont divisées en 3 trois ensembles. Le premier ensemble SE de 9 vues qui sont encodées par Joint Exploration Model (JEM), le deuxième ensemble SR contient les 7 vues adjacentes des vues SE et elles sont approximées linéairement [ZC17] et le troisième ensemble SI représente les vues manquantes à synthétiser. Ce nouveau schéma est basé sur CNN [KWR16] et on a apporté trois améliorations différentes, chacune donne un gain en BD-rate et BD-PSNR par rapport aux autres méthodes de l'état de l'art:

1. Réglage de la qualité de la vue centrale (VC) de l'image LF est codée comme un frame intra. Elle est utilisée par les blocs de l'apprentissage linéaire (LA) et CNN comme référence pour la prédiction de toutes les autres images. Ainsi, la qualité

de cette VC est un facteur clé pour la prédiction et la génération d'autres vues. Pour une VC de haute qualité, on lui attribue un paramètre de quantification Qintra tel que: Qintra= Q + Qoffset où Qoffset =-4 (fixé de manière empirique).

2. Optimisation débit-distorsion Pour reconstruire les vues intermédiaires (SI), nous avons proposé d'effectuer un RDO pour chaque vue intermédiaire, indiquant ainsi quelle méthode entre LA et CNN peut fournir les performances de RD les plus élevées en calculant la fonction de coût J (J = D + $\lambda$.R, $\lambda$ est le multiplicateur de Lagrange).
   On a constaté que la valeur $\lambda$= 0.1 est optimale et que l'optimisation lagrangienne donne les meilleures performances. On a sélectionné ensuite la meilleure approche en minimisant le coût de RD (J) pour chaque vue intermédiaire.

3. Post-traitement Enfin et comme un post-traitement, nous introduisons un processus de correspondance superpixel en pixel pour améliorer encore plus la qualité des vues approximées et synthétisées [DSS17].

Dans le quatrième chapitre, nous faisons une évaluation subjective et objective pour les méthodes de compression de l'état de l'art, sur un ensemble d'images LF de plusieurs bases de données différentes. A ce titre, nous décrivons en détail les conditions des tests psychovisuels ainsi que les normes de l'Union Internationale des Télécommunications (UIT) qui y sont associées [BT.12b]. Lors du choix des images, nous avons pris en compte les trois facteurs: information spatiale (SI), colorfulness (CF) et le nombre de pixels occultés [P.908, DH03, WER16].
Les tests se sont déroulés dans la salle psycho visuelle du laboratoire IETR-Rennes en 2 phases et avec des conditions d'éclairage conformes à la recommandation ITU-R BT.500. 18 observateurs ont fait ce test en utilisant 4 débits pour les pseudo-vidéos avec 9 frames per second (fps). En particulier, sur un seul écran on affiche 2 pseudo vidéos (l'original à gauche et la vidéo codée à droite). L'observateur choisit donc un score entre 1 et 5, suivant l'échelle suivante: 1) Très gênante, 2) Gênante, 3) Légèrement gênante, 4) Perceptible mais pas gênante, et 5) Imperceptible. En effet, certaines données peuvent biaiser les résultats. Ainsi, un processus de filtrage a été appliqué sur les données de l'expérience en se basant sur la recommandation ITU-R BT.500.
Globalement, les méthodes de codage LF basées sur la synthèse de vues (qui sont basées sur l'apprentissage linéaire ou bien Deep Learning) offrent la meilleure qualité visuelle à tous les débits, par exemple, pour la plupart des images LF, leur qualité visuelle fournie à un débit moyen est à peu près identique à celle obtenue par les approches de codage classique 2D à haut débit. Ainsi, les méthodes de codage basées sur la synthèse de vues peuvent atteindre des performances de codage élevées et démontrer leur efficacité en fournissant la meilleure qualité visuelle par rapport aux deux autres méthodes.

Dans le cinquième chapitre, nous utilisons le D2GAN est un type de l'architecture du Generative Adversarial Network (GAN) afin d'avoir un seul générateur avec 2 discriminateurs. L'idée est inspirée de [NLVP17] qui mentionne qu'une telle architecture donne plus de stabilité au Generative Adversarial Network (GAN) avec de meilleurs résultats.

Dans cette architecture, le générateur se compose de deux réseaux de nuerons le premier pour estimer la disparité (qui est légèrement différente) et le deuxième pour estimer les couleurs. Le premier discriminateur a toujours le même rôle, c.à.d. donner un score élevée si l'image est réelle. Alors que le deuxième discriminateur donne un score élevé si l'image est générée par le générateur.
Il faut noter, que chacun des discriminateurs possède une fonction perte distincte avec une configuration paramétrique distincte. Nous allons appliquer D2GAN sur les vues de références encodées par le standard VVC.
Nous proposons une méthode de compression basée sur D2GAN et l'encodeur Versatile Video Coding VVC. il s'agit de l'intégrer avec le schéma hiérarchique de VVC. L'idée globale consiste à éviter de coder les vues du niveau hiérarchique supérieur et les générer avec D2GAN au niveau du décodeur. Une extension pour cette approche est envisageable, et consiste à faire une optimisation pour décider de coder ces vues du niveau supérieur avec VVC ou de les générer par D2GAN, et on aura un schéma VVC hiérarchique avec Rate Distortion Optimization (RDO) et D2GAN.

# Introduction

Computer vision applications such as refocusing, segmentation and classification become one of the most advanced services in the field of image processing. Such applications require rich semantic information from the scene. 3D technology is widely used in the fields of entertainment, education and medical imaging. There are different ways to represent 3D information. One of the most common is to associate with a classic 2D image called texture, an image of Depth of Field (DoF). This joint representation thus allows for a good 3D reconstruction as soon as the two images are well correlated, and more particularly for the contour areas of the depth image. In comparison with classic 2D images, knowledge of depth of field for 3D images provides therefore provides important semantic information about the composition of the scene.

Figure 3: The LF technology allows a very faithful reproduction of reality in Virtual Reality (VR)

Another technology that is gaining more importance is LF technology. The LF image is an unconventional image, containing much more information than the intensity of the light rays that interact with the scene. It gives a very rich description of a 3D scene allowing to offer a wide range of functionalities. For example, it allows the advanced synthesis of intermediate views, as well as the re-focusing of the image after acquisition. Interest in Light Field LF technology continues to grow very strongly, in particular,

with the increasing penetration of acquisition and display devices for LF content in the consumer market.

In particular, a dense set of cameras and microlens arrays are used as the Plenoptic (Figure 4), to have the direction of every ray coming from the stage to the LF acquisition system [NLB+05] . This can be extracted and represented by spatial and angular coordinates. However, such an imaging system has many disadvantages, including the large amount of data produced and the complexity increase for the representation of the scene. This therefore raises the urgent question of their compression.


(a) Lytro Illum (b) Raytrix

Figure 4: Samples of Light Field camera.

## Contributions

In this thesis, we first propose a CNN-based LF coding scheme that includes RDO followed by post-processing. The main concept is to exploit the correlation between the different LF views and avoid the coding of all the views. The LF views are thus divided into 3 sets: a first set which is coded by a standard 2D encoder, a second set which is linearly approximated and a third one set which will be synthesized at the decoder either by linear approximation or by synthesis with CNN according to the decision made by the RDO block. Next, we integrate the D2GAN with the VVC hierarchical encoder. The overall idea is to avoid coding the views of the higher hierarchical level and generate them with D2GAN at the decoder level. Finally, we evaluate the scheme proposed subjectively on a set of LF images of several different databases. The conditions of psychovisual tests comply with the standards of the International Telecommunication Union (ITU). The most notable feature is that view-based coding methods can achieve high coding performance and demonstrate their effectiveness by providing the best visual quality over the other two methods.

## Outline

This manuscript is organized as follows:

**In Chapiter 1,** we make a state of the art of the characteristics of the LF image and the existing compression methods in the literature. The LF image is an unconventional

image, containing information much more than the intensity on the rays that interact with the scene. The LF image gives a very rich description a 3D scene allowing to offer a wide range of functionalities. In particular, it allows the advanced synthesis of intermediate views, as well as the re-focusing of the image after acquisition. There are two LF image acquisition systems, generating different types of LF representations. In this chapter, we describe the whole chain from LF acquisition to LF visualization, along with the different functionalities it offers.

**In Chapiter 2,** we analyze the different existing LF image coding techniques that apply to the various representations of the light field (e.g. LF raw image, sub apertures, epipolar). Several approaches are adopted by the state-of-the-art, such as pseudo-sequence or predictive coding, while presenting them in details along with their performance.

**In Chapiter 3,** we explain our proposed new LF image coding scheme. In this scheme, we consider the LF image as 8*8 subapertures views. The views are divided into 3 three sets. A first set of reference views are encoded with a standard encoding method. Then, the other set of views are either linearly approximated or synthesized using CNN based on a rate distortion optimization. Finally, our proposed method applies a post processing on a block level for further quality enhancement.

**In Chapitre 4,** we make a subjective and objective assessment for state of the art compression methods, on a set of LF images from several different databases. As such, we describe in detail the conditions of psycho-visual tests defined by as well as the ITU standards. For the images choice, we took into account the three factors: spatial information, colorfulness and the number of pixels occluded. 18 observers performed this test. Overall, LF coding methods based on view synthesis (which are based on Linear Approximation (LA) or Deep Learning (DL) offer the best visual quality at all bitrates. For example, for most LF images, their visual quality provided at an average bitrate is about the best as that obtained by conventional 2D high-speed coding approaches.

**Chapiter 5** presents another contribution for LF image coding. We use the D2GAN which is a specific type of architecture of the GAN, with a single generator with 2 discriminators. The idea is inspired by [NLVP17] where they state that such an architecture gives GAN more stability with better results.

In this architecture, the first discriminator has always the same role, i.e. to give a high score if the image is real, while the second discriminator gives a high score if the image is then generated by the generator. It should be noted that each of the discriminators has a distinct loss function with a distinct parametric congregation. We will apply D2GAN to reference views encoded by the VVC standard. We offer a compression method based on D2GAN and the VVC encoder. It consists of integrating it with the VVC hierarchical scheme. The overall idea is to avoid coding views from the higher hierarchy level and generate them with D2GAN at the decoder level. An extension

for this approach is possible, and consists in making an optimization to decide to code these upper level views with VVC or generate them by D2GAN, and we will have a hierarchical VVC scheme with RDO and D2GAN.

# Chapter 1

# Light Field Technology

## 1.1 Introduction

Images are invading the world internet traffic. In fact, from the very basic Black and White (BW) images to the most recent 3D images, in addition to the need to memorize the moment and relive it is highly increasing. LF proposes a new approach for the acquisition of scenes. The spectral images record the information of the light rays propagating in the scene including their directions and intensity. The amount of information allows to render different views with varying DoF and focal planes without the need of re-acquiring the scene. Therefore, LF imaging systems is becoming one of the most popular techniques for Virtual Reality, Augmented Reality (VR/AR), Teleconferencing, and E-learning.

In this chapter, we describe in detail all aspects of LF image processing as shown in Figure 1.1. In Sections 1.2 and 1.3, we present the LF acquisition and representation, while in Section 1.4, we briefly discuss the different LF compression techniques. Moreover, the LF visualization is explained in Section 1.5. Section 1.6 describes its important functionalities. As for section 1.7, it presents the different LF display technologies. Finally, Section 1.8 concludes this chapter.

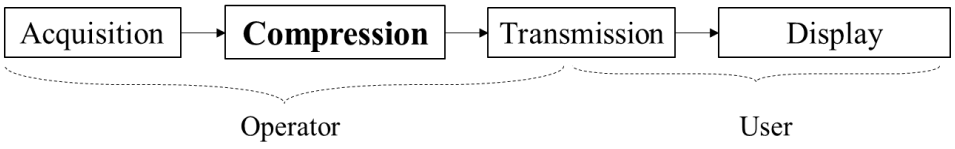



Figure 1.1: LF imaging processing flow [PdSL+17]

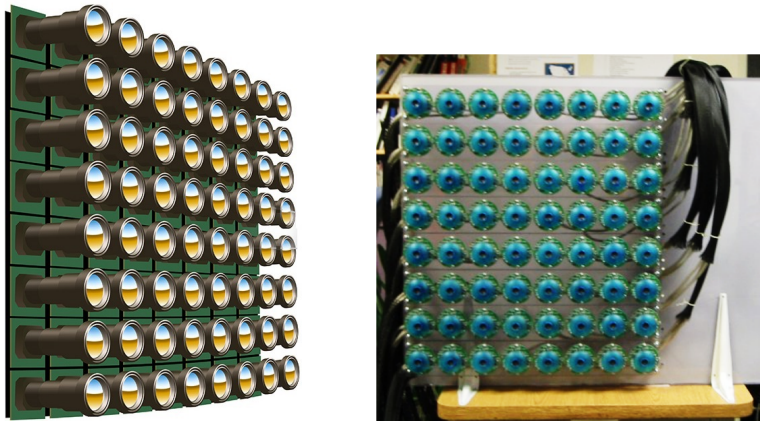
## 1.2 LF Acquisition

Unlike conventional image, the Light Field image records both spatial and angular light radiance in one shot. The acquisition of Light Field image can be obtained by

using a camera array, one mobile hand camera, or a plenoptic camera using arrays of micro-lenses placed in front of the photosensor, generating LF images with small baselines [NLB+05]. In general, different acquisition techniques can be used to capture Light Field images depending on the requirements for baseline (i.e., the physical space that will be covered by the $(U, V)$ sampling), and on the image resolution. In the following subsections, we describe in details the two LF acquisition technologies.

### 1.2.1 Camera Array

A camera array consists of many traditional cameras organized in horizontal and vertical alignments at regular baselines to capture the same view from different viewpoints, as illustrated in Figure 1.2a. In this case, the cameras synchronization, their color and geometrical calibrations are considered as immense technical challenges. Moreover, they create large data volume and often high energy consumption. For these reasons, camera arrays are still quite rare, but there are few notable designs. For example, Stanford Multi-Camera Array [WJV+05] records LF images (see Figure 1.2a) with almost 6000 pixels wide [YLX16], which is too high in comparison with the High Definition video cameras that provide a resolution of 1920×1080 pixels. In this case, the $(U, V)$ sampling depends on the baseline parameters of the camera array grid. The full 4D LF is formed and new views corresponding to narrower baseline parameters must be further synthesized, if needed. An example of such acquisition technology is the Stanford Multi-Camera Array.



(a) Stanford's multi-camera array

(b) Distributed Light Field camera

Figure 1.2: Examples of LF camera arrays: (a) Stanford's multi-camera array, in which conventional cameras are arranged regularly in a linear array with full parallax [RMS16] and (b) distributed LF camera, 64 cameras with distibuted rendering [YEBM02].

To overcome data bandwidth problems, in 2002, Yang et al. [YEBM02] used 8×8 video cameras in a proper design to capture dynamic Light Field as illustrated in Figure 1.2b, and employed a distributed rendering algorithm.

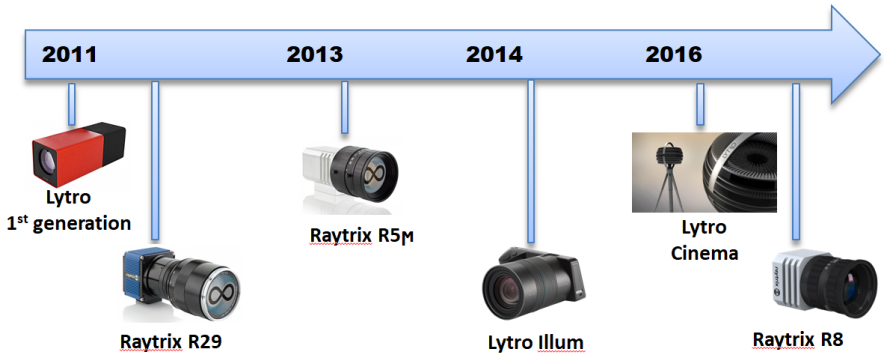



Figure 1.3: Timeline of the plenoptic cameras announced recently in the market [TZAG13, RAY]

### 1.2.2 Plenoptic Camera

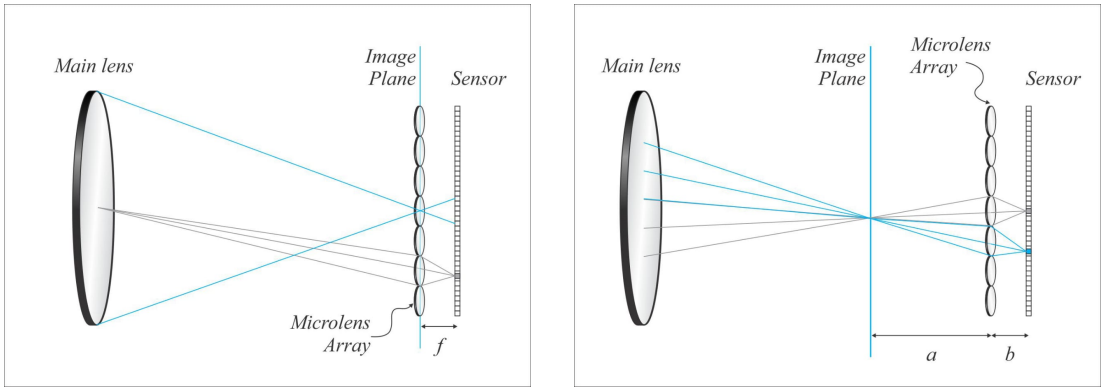
For LF image acquisition with narrow base, a single lens stereo camera can be used. It consists of a hand-held plenoptic camera with added optical elements in front of the sensor. Alternatively, a plenoptic camera can, in a single photographic plane, record the Light Field on its imaging plane. The camera largely resembles looks like a regular digital camera, operating similarly though recording LFs instead of regular photographs. Figure 1.3 shows the Lytro[1] [TZAG13] and the Raytrix[2] plenoptic cameras. Currently, plenoptic cameras are commercially available in the market from two sources: the Lytro camera based on the "plenoptic 1.0" (recently acquired by Google) targets ordinary consumers, and the Raytrix- based on the "plenoptic 2.0" -targets industrial applications as illustrated in Figure 1.4.

The used technique is called integral photography. It is widely used in several imaging fields including engineering, optics and the study of animal vision. It consists of using an array of microlenses inserted in front of the photosensor in a conventional camera. The size of microlenses is microscopic when compared with that of the main lenses, and so is the gap between the microlenses and the photosensor. Covering multiple photosensor pixels, each microlens separates the light rays that hit it into a minute image on the pixels underneath.

Figure 1.5 shows an over simplified 2D plenoptic camera with 2 microlenses and 3 pixels, where the main lens plane represents the angular plane and the microlens plane represents the spatial one. Therefore, each pixel in a microlens image corresponds to the same scene point. Conversely, corresponding pixels between two microlens images correspond to two different scene points imaged at the same angle.

In this kind of camera, the imaging plane is that of the microlens, which sets the spatial sampling resolution with its size. In Figure 1.7, a grid of boxes lying over the ray-space diagram outlines the sampling of the Light Field recorded by the photosensor pixels. Each of these boxes denotes the cluster of rays contributing to one pixel on the

[1] http://www.lytro.com/imaging
[2] https://www.raytrix.de/




(a) Plenoptic camera 1.0 (b) Plenoptic camera 2.0

Figure 1.4: Plenoptic 1.0 and Plenoptic 2.0 cameras [APKS18] optical design also called unfocused and focused plenoptic camera [LG09] respectively. The fundamental difference in the optical setup between Plenoptic 2.0 and Plenoptic 1.0 is that in the former, the micro-images are focused on the scene (through the relay system), while in the latter they are completely defocused relative to the scene.

Table 1.1: A summary of typical Light Field acquisition approaches.

| Approach | Year | Implementation | Resolution | Capture speed |
|---|---|---|---|---|
| Yang et al. [YEBM02] | 2002 | 8×8 camera array | 320×240×8×8 | 15-20 fps |
| Wilburn et al. [WJV+05] | 2005 | 10×10 camera array | 640×480×10×10 | 30 fps |
| Light Field gantry [Ada02] | 2002 | Gantry | 1300×1030×62×56 | 5h/slab |
| Ng et al. [NLB+05]] | 2005 | Microlens array | 292×292×14×14 | 16 ms |
| Lytro Illum [TZAG13] | 2014 | Microlens array | 625×434×15×15 | 3 fps |

photosensor. Rays were marked from the borders of each photosensor pixel out into the world through its parent microlens array and the glass elements of the main lens so as to measure the sampling grid.

## 1.3 LF Representation

In general, LF is represented as a vector function that describes the location, the direction and the intensity of each ray of light within the scene. There are several ways to represent the scene in LF imaging that we are explained in the following subsections.

### 1.3.1 Plenoptic Function

The plenoptic function was first introduced by Adelson and Bergen [AB91]. It can be described by a 7 dimensional function as follows:

$$L(\lambda, t, x, y, z, \theta, \phi), \tag{1.1}$$

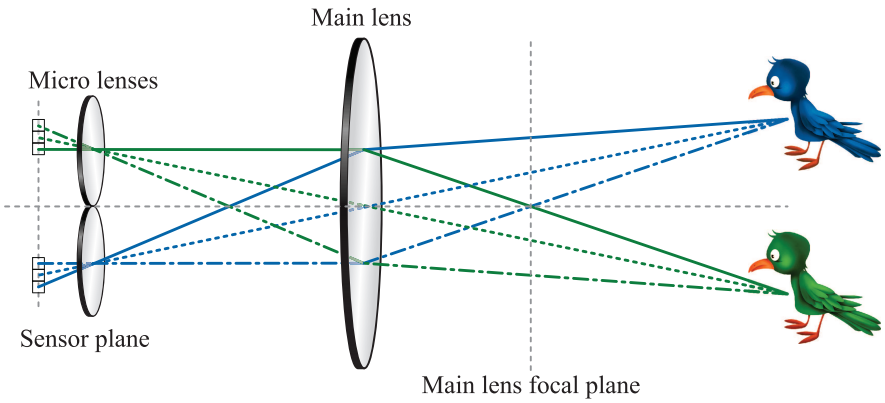



Figure 1.5: The lenslet-based plenoptic camera. Plenoptic camera with 2 microlenses and 3 pixels. Here the angular plane corresponds to the main lens plane and the spatial plane to the microlens plane [HSG17].

where $(x, y, z)$ are the spatial coordinates, $(\theta, \phi)$ are the angular coordinates, $\lambda$ is the wavelength and $t$ is the time. The plenoptic function $L$ assigns to every point in free space and to every direction a corresponding radiance for specific wavelength $\lambda$ and time as shown in Figure 1.6.

For static scenes, the dimension of this function is reduced to 5 dimensions without considering time and wavelength [LH96]. Further, assuming that the rays are passing in un-occluded pixels, one can simplify it as a 4D LF as shown in Figure 1.6 where $(u, v)$ plane represents the microlens plane and $(s, t)$ represents the spatial one.

### 1.3.2 The Lumigraph

In this representation, the Light Field signal $L(s, t, u, v)$ describes all light rays passing through the $(s, t)$ and $(u, v)$ planes called the lumigraph [GGSC96]. The points of intersection of a ray with two parallel planes completely describes its position and orientation in the free space. By convention, the $(s, t)$ plane is close to the camera, and the $(u, v)$ plane is close to the scene. The two-planes parameterization describes rays in terms of position and direction, and so the terms angular and spatial are sometimes employed to describe these dimensions. One interpretation is that $s$ and $t$ defines the position of a ray, while $u$ and $v$ defines the direction.

## 1.4 LF Compression

Acquiring LF images creates a vast amount of data: around 150 MB for lenslet images with 15×15 viewpoints of resolution of 635×434, around 6.8 GB for 15×15 4K images acquired with a multi-camera array. This large information is the first challenge represented by the large amount of data in addition to increasing the complexity of representation in the scene and redundancy of information. Therefore, one need to find

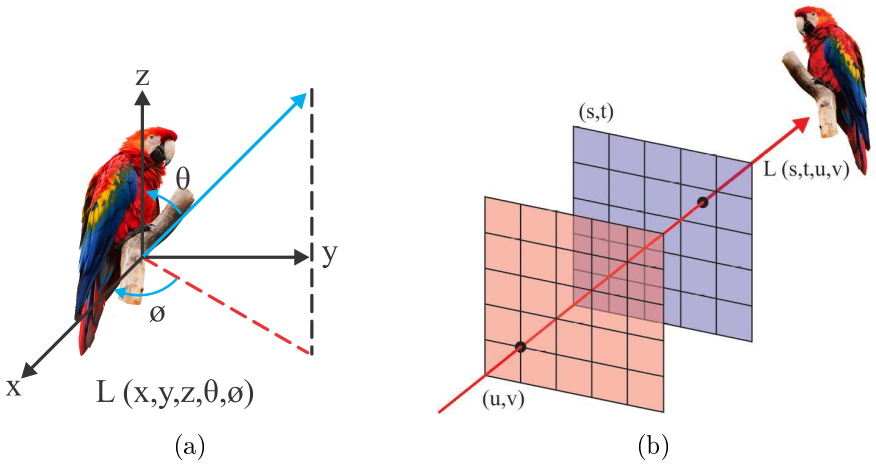



Figure 1.6: (a) Spatial parametrization of 5D-LF representation and (b) 4D-LF representation.

efficient methods to compress the LF images. Different LF compression techniques will be detailed in the next chapter.

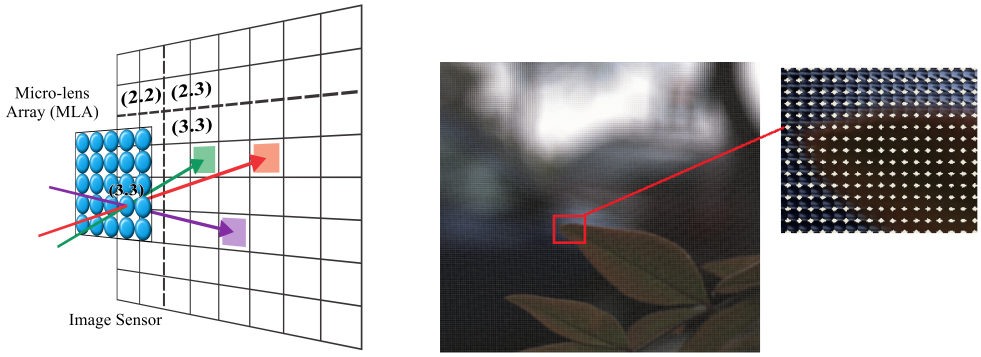
## 1.5 LF Visualization

Although the function $L(s, t, u, v)$ is a simplified Light Field model, it is still hard to imagine this 4D representation. Thus, there are several representations of the LF image including micro-image, sub-aperture and epipolar image as illustrated in Figures 1.7, 1.8, 1.9 respectively and explained in the following subsections.

### 1.5.1 Lenslet Images

Photosensor pixels are assigned to each microlens and form a small image. This image is referred to as the microlens image. In the raw Light Field photograph, there are as many microlens images as the number of microlenses.

Each microlens image shows the incident light ray that leaves from different positions and arrives at the photosensor through the microlens array [LZM09]. A certain point on the $(u, v)$ plane represents the light rays bound of all points on the $(s, t)$ plane (the collection of light rays from different viewpoints projected onto a certain point, i.e., the same point as seen from different viewpoints, see Figure 1.7). The lenslet image corresponds to a set of micro-images and can be saved as png file (demosaiced raw images). The width and height of raw lenslet image captured by the Lytro Illum are 7728 and 5368, respectively.

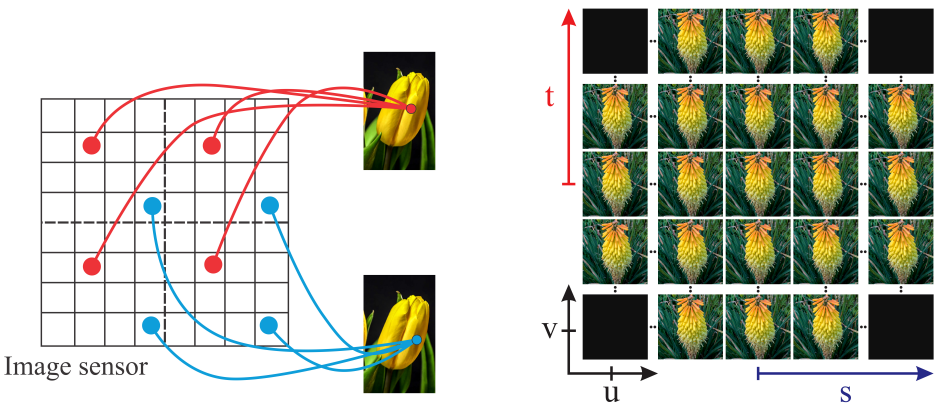



(a) (b)

Figure 1.7: (a) Rays captured by the microlens (3,3) and the process passing to the photosensor and (b) Raw Light Field photograph.

### 1.5.2 Sub-aperture Image

Sub-aperture images are made by reordering incident rays in the raw Light Field photograph. Each sub-aperture image is composed of the pixels of same position selected from each microlens image.




(a) (b)

Figure 1.8: (a) Process of making a sub-aperture image and (b) Sub-aperture images from a plenoptic camera.

Thus, the 4D LF can be represented as a 2D array of images with a smaller baseline,

such as the one shown in Figure 1.8 (a). The lenslet image can render in form of multi-view sub-aperture images by putting together the pixels in the same position within each micro-image to create a rendered image for a specific viewpoint, (see Figure 1.8 (b)).

The extraction process implies that the number of sub-aperture views amounts to the number of pixels in micro image (see Figure 1.8). Consequently, the effective resolution of a sub-aperture image equals the number of micro lenses in the plenoptic camera.

### 1.5.3 Epipolar Image

The epipolar plane image (EPI) is obtained by fixing the coordinates in both the spatial and angular dimension. The large $(u, v)$ slice can be thought of as a conventional image taken from a camera sitting on the $(s, t)$ plane. Each epipolar image is the 2D slice of the Light Field where $t$ and $v$ are fixed, and $s$ and $u$ vary. The $(s, u)$ and $(t, v)$ slices are sometimes referred to as epipolar images (see Figure 1.9).

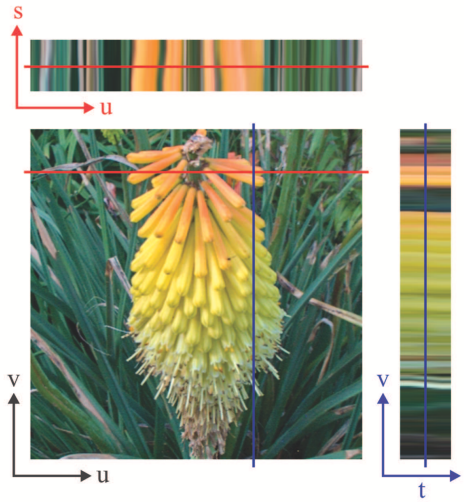



Figure 1.9: The epipolar plane image

In our contribution, we will adopt the sub-aperture representation.

### 1.5.4 Public LF Dataset

This section presents important open source Light Field images datasets. The properties of these datasets are summarized in Table 1.2, including synthetic, real-world and microscopy LF scenes.

Table 1.2: Most relevant datasets with corresponding features.

| Dataset | Year | Features | Acquisition devices |
|---|---|---|---|
| Stanford LF archive | 2008 | More than 20 images | Microscope and Camera Array. |
| Synthetic LF archive | 2013 | more than 17 Light Field images, includes transparencies, occlusions and reflections. | Camera (Artificial LF) |
| EPFL LF image dataset | 2015 | More than 118 images with different categories: urban, landscapes, etc. | Lytro Illum |
| INRIA LF dataset | 2017 | More than 46 images, low lighting conditions, indoor and outdoor. | Lytro Illum |
| Our LF dataset | 2018 | More than 30 images, indoor, outdoor transparencies, occlusions and reflections. | Lytro Illum |

## 1.6 LF Functionalities

As previously introduced, the flow of rays captured by Light Field acquisition devices is in the form of large volumes of data retaining both spatial and angular information of a scene. Thus, from a single exposure, it enables a variety of post-capture processing capabilities such as: re-focusing, extended focus, changing the point of view and depth estimation. In the following, we explain in details some of these advanced functionalities.

**Re-focusing:** Blurred zones or regions in a 2D image are caused by scattered rays received by the sensors of a 2D standard camera. One way to get theses regions in focus is to capture the scene from different perspectives. As a consequence, the angular information of light rays is acquired. Such a job can be achieved using plenoptic cameras as mentioned in Section 1.2. Therefore, LF technology allows to generate refocused image by using multiple techniques such as Fourier transform [NLB+05] and It simply relies on the LF Toolbox software [Dan14] that is developed by *D. Dansereau*.It, mainly, uses function *LFFiltShiftSum*. This works by shifting all the available sub-aperture images of each Light Field image to the same depth, and then adding all the sub-aperture images together to produce a 2D depth plane extracted from the original Light Field.

Figure 1.10: LF image refocusing: (left) refocused on foreground and (right) refocused on background

This function uses an input value called slope, which allows controlling the optical focal plane, and the object that should be focused. The relationship between slope and depth depends on LF parameterization, but in general a slope of 0 lies near the center of the captured depth of field. When the image is digitally refocused on the background, images in the foreground may appear ghosted and vice versa (see Figure 1.10). It is known that the Light Field refocusing operation has denoising properties [DBPW13], thus refocusing applied on compressed sub-aperture images (SAI) will reduce the distortion due to compression artefacts.

**Changing viewpoint:** The huge information in the LF image provides the ability to see the scene from different viewpoints. LF was constructed from rendered images of a buddha computer model.

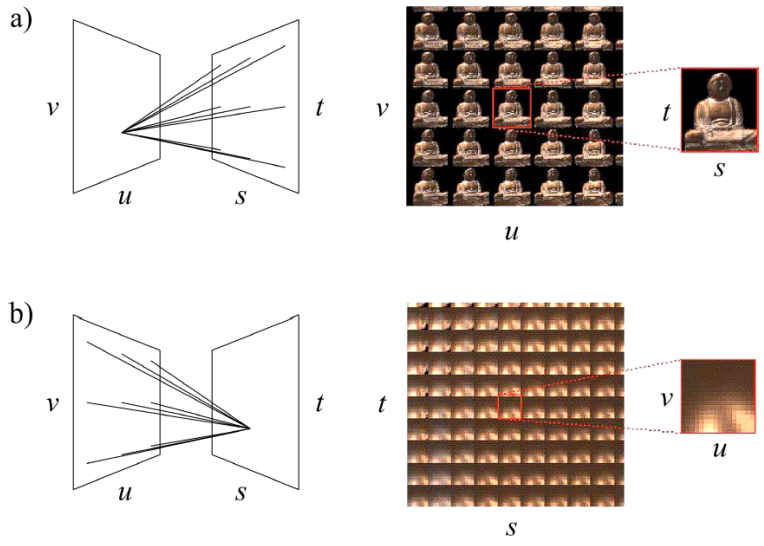



Figure 1.11: Actually seeing two visualizations here, (a) an st-arrays of $(u,v)$ images or (b) uv-arrays of $(s,t)$ images. First on (a), each image is the angular distribution of rays around a point on the $(u,v)$ or camera plane. It looks like a perspective view of the scene. On (b), angular distributions around points on the $(s,t)$ or focal plane. These look reflectance maps because the object is near the $(s,t)$ plane [LH96].

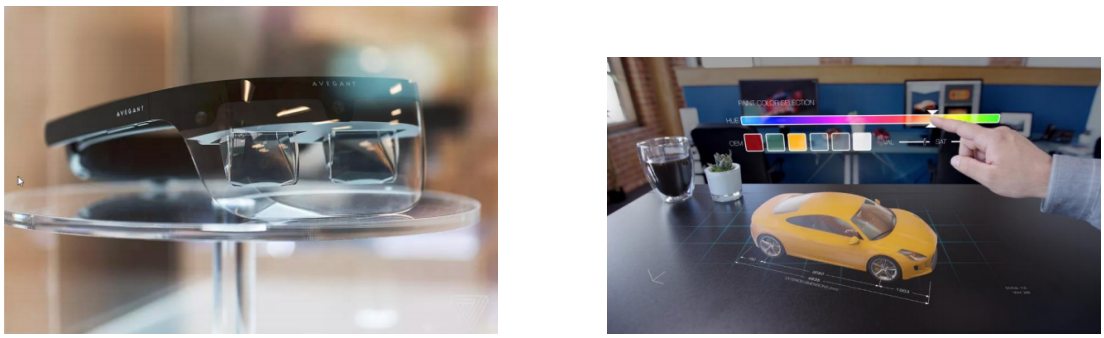
We can zoom in and roam around. At each observer viewpoint, the view with correct perspective and shading is computed by extracting a two-dimensional slice from the 4D-LF parametrization as shown in Figure 1.11.

## 1.7 LF Display

As introduced, due to the 4D representation of LF rays, LF technology allows to replicate real-world scenes with advanced features such as 360 video, refocusing, VR/AR. In this

context, among the possible display technologies that are currently available for LF content visualization, one can cite:

- 2D displays: In this case, a single 2D view or more specifically a 2D version of the LF content has to be rendered from the decoded LF content.
- Stereo displays: In this case, a pair of views need to be rendered from the LF image and delivered to the display. This type of display technology improves the users' depth perception (with respect to the 2D display) by presenting a different view to their left and right eyes (typically, by means of a pair of eye-glasses).
- Multiview auto-stereoscopic displays: Multiview autostereoscopic is a glassless display technology that allows creating a more natural 3D illusion (with respect to the stereo display) to the end-user. It presents a different perspective as the user moves horizontally around the display (known as horizontal motion parallax). In this case, multiple views need to be rendered from the LF content and delivered to the display. Moreover, following the recent developments in sensor and optical manufacturing, the display technologies are also evolving for providing a more natural and immersive visualization. Therefore, some prospective display technologies started to emerge. Among them, it is possible to cite:
- Super-multiview LF displays, as proposed by Holografika [KY18, LPT16] which uses a very dense number of views to create a replica of the 4D LF.



(a) (b)

Figure 1.12: (a) Interactive mixed reality head-mounted viewer and (b) using a head-mounted viewer, one can visualize and make design changes in real time.

- Augmented Reality (AR) and VR displays: One of the newest technologies, that is gaining momentum in the past few years, is the development of commercially available Light Field Displays. These displays project synthetically rendered light rays with the necessary depth and colour cues. The radiance image is a pixel representation of the Light Field, where every pixel represents the position and orientation of a light ray passing through the display surface [DBPW19]. The light rays are then angularly distributed by a microlens array that is not affected by the

viewer position. The result is naturally rendered holographic objects right in front of the viewer. This solves one of mixed reality's greatest technical challenges: Enabling the virtual holographic objects to appear real from different angels [CC15]. Two main implementations of the new technology are by integrating it in: 1) interactive head-mounted viewers and 2) table top displays.
With head mounted mixed-reality viewers, the built-in Light Field displays send images with multiple focal points to the retina, mimicking the way of the light in the real world reflects off objects to hit a person's eyes, as illustrated in Figure 1.12a. Architects and designers, for example, are using such viewers to transform traditional 2D models and sketches into 3D holographic assets visualized in the real world accurately displayed across all focal distances as shown in Figure 1.12b.
With table top displays, the synthetic Light Field computed from a 3D model is projected through an array of microlenses to create a 3D aerial holographic scene for all viewers, as illustrated in Figure 1.13a.

Car manufacturers are working on integrating LF displays in their upcoming vehicles to create innovative cockpit solutions. The Light Field displays enable information to be safely presented to the driver in real-time, allowing the driver's interaction with the vehicle to become more comfortable and intuitive. It also allows passengers in the front and back seats to share the 3D holographic experience with the driver 1.13b.


(a) (b)

Figure 1.13: (a) Array of microlenses responsible for the angular distribution of light rays and (b) Light Field displays used to render 3D holographic information for drivers.

Technology firms such as the Light Field Lab have raised millions of dollars recently in funding to advance the LF display technology and produce large 3D holographic live scenes for large venues. Viewers can soon enjoy the fully interactive, social experiences with their friends without the need of specialized headsets.

Figure 1.14: The future of LF displays. Interactive 3D holographic scenes in large venues.

## 1.8 Conclusion

In this chapter, we have provided an overview of Light Field image definition, acquisition and some of the types of cameras used to capture it. Among them we focused on the Lytro Illum Camera that enables the acquisition of a baseline LF images. Moreover, we presented the different ways to represent the Light Field content. We are more interested in sub-aperture views representation used in our compression LF scheme. Then, the main features, including refocusing and changing the viewpoint of the scene were described. Finally, the most important recent techniques for displaying the LF image were presented. In the next chapter, we will focus on the state-of-the-art LF image coding solutions.

# Chapter 2

# Light Field Image and Video Coding: a Review of the Literature

## 2.1 Introduction

In this chapter, some related concepts to the Light Field image are introduced. Then, multiple approaches that are proposed for Light Field image coding will be analyzed. Furthermore, this chapter will present the main concepts of machine learning and finally will explain the different techniques for Light Field visual quality evaluation detailing the testing considered environment requirements.

## 2.2 Related Concepts

This section, will explain some of the definitions used in this research, such as the whole family of norm to measure the vector's magnitude and superpixel segmentation [ASS+12].

### Vector Norm

Firstly, we assume a vector $\vec{v}$ as an ordered tuple of numbers.

$$\vec{v} = (v_1, v_2, ..., v_n), (v_i \in \mathbb{R}, for\ i = 1, 2, 3, ..., n) \tag{2.1}$$

L1-norm: the L1-norm [Wei12] (also known as $\ell 1$-norm, or mean norm) of a vector $\vec{v}$ is denoted $\|\vec{v}\|_1$ and is defined as the sum of the absolute values of its components:

$$\|\vec{v}\|_1 = \sum_{i=1}^{n} |v_i| \tag{2.2}$$

L2-norm: the L2-norm (also known as the $\ell 2$-norm) of a vector $\vec{v}$ is denoted $\|\vec{v}\|_2$

and is defined as the square root of the sum of the squared vector values.

$$\|\vec{v}\|_2 = \sum_{i=1}^{n} \sqrt{{v_i}^2} \tag{2.3}$$

Figure 2.1: Images segmented using Simple Linear Iterative Clustering (SLIC) into superpixels of size 64, 256, and 1.024 pixels (approximately)

L infinity-norm: the infinity norm (also known as the L$\infty$-norm, $\infty$-norm, max norm, or uniform norm) of a vector $\vec{v}$ is denoted $\|\vec{v}\|_\infty$ and is defined as the maximum of the absolute values of its components:

$$\|\vec{v}\|_\infty = max\{|v_i| \ : \ for\ i = 1, 2, 3, ..., n\} \tag{2.4}$$

### Superpixel Segmentation

The Image segmentation is referred to as one of the most important processes of image processing. Image segmentation is the partition of an image into regions or categories (sets of pixels, also known as super-pixels), which correspond to different objects or parts of objects. Every pixel in an image is allocated to one of a number of these segments [PP93]. A simple technique of segmentation consists of using the gradient. A superpixel can be defined as a group of pixels that share common characteristics (such as pixel intensity).

SLIC is a particular type of segmentation, where pixels are grouped into perceptually meaningful atomic regions as shown in the Figure 2.1. It is mainly used to compute image features, and greatly reduces the complexity of subsequent image processing

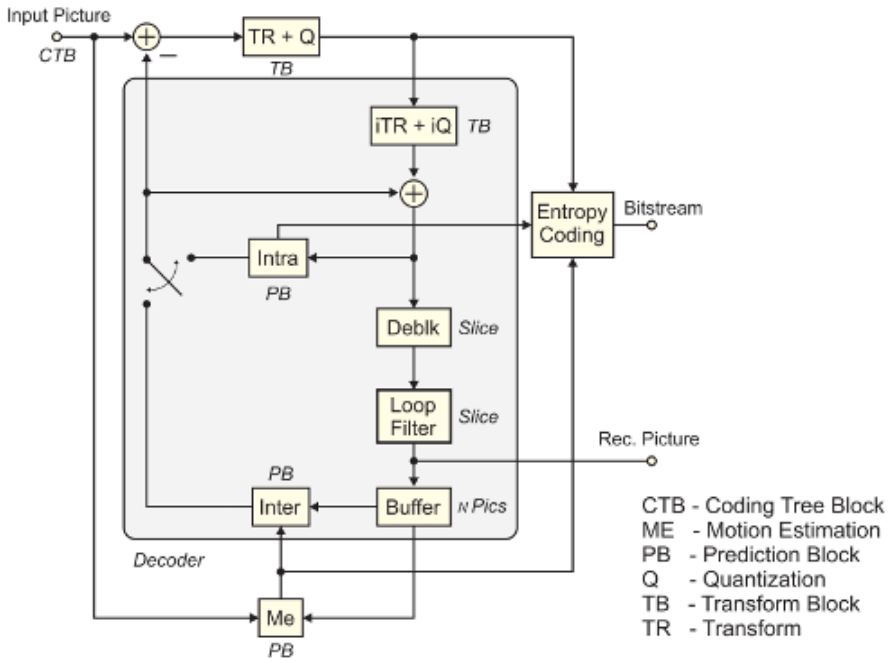



Figure 2.2: Illustration of the hybrid video coding scheme

tasks. This research uses the SLIC superpixel algorithm that performs a fast and efficient semantic atomic segmentation [ASS+12]. SLIC generates superpixel regions by adopting the k-means approach with two important distinctions.

- The number of distance calculations in the optimization is dramatically reduced by limiting the search space to a region proportional to the superpixel size. This reduces the complexity to be linear according to the number of pixels $N$ and independent from the number of superpixels $k$.
- A weighted distance measure combines color and spatial proximity while simultaneously providing control over the size and compactness of the superpixels.

## 2.3 Principle of Current Video Compression Standards

Before the introduction specific light filed coding schemes, it is necessary to present and analyze the principal of current video coding. This section introduces the main standards for 2D image and video compression. The first problem is the huge bandwidth needed for transmitting such huge image/video data. To reduce storage requirements and improve transmission bandwidth, redundancies within image and video signals can be exploited to compress the content more efficiently. The focus is on the two last generation video coding standards, HEVC and VVC, that are used for this work. These

latter integrated a set of new coding tools, extending the existing hybrid coding concept as illustrated in Figure 2.2 based on prediction, residual error transformation and quantization.

### 2.3.1 Redundancies removal

A video consists of a succession of frames. Each individual frame can be viewed as an individual static image. Therefore, multiple frames may share some common properties or features called redundancies. Different types of redundancies can be found in a video.

Spatial redundancy: pixels or regions that are duplicated within the same frame.

Statistics redundancy: in order to store the pixels of the image, the coding information (modes, coefficients, etc...) are described as a succession of symbols (or set of bits). The distribution of these symbols is not random, then some correlations can be exploited by source coding like arithmetic coding algorithms. Remaining statistical redundancies can be further exploited by using entropy coding such as Context-Adaptive Binary Arithmetic Coding (CABAC).

Temporal redundancy: The exiting correlations within two consecutive frames in the video.

Exploiting spatial, temporal and statistical redundancies is one of the primary techniques in video compression.

### 2.3.2 High Efficiency Video Coding

This section gives a brief overview of the state-of-the-art HEVC/H.265 standard [SHDP17, SOHW12]. The HEVC standard, or H.265/Motion Picture Expert Group (MPEG)-H Part 2, is finalized by the Joint Collaborative Team on Video Coding (JCT-VC) in 2013. HEVC was designed to bring a bit-rate reduction of 50% compared to its predecessor, the Advanced Video Coding (AVC)/H.264 codec [WSBL03]. The aim of this section is to explain in details some of these advanced features.

The video sequence is first organized into multiple Group of Picturess (GOPs) of a fixed number of consecutive frames. The GOP structure defines the encoding order of the frames. A classical GOP structure in HEVC is the hierarchical GOP structure, called Random Access (RA) coding configuration. The first frame in the GOP is encoded independently as an Intra (I)-frame (using only Intra predictions), the last frame as a Predicted (P)-frame (predicted from the first frame or other past frames from past GOPs as shown in Figure 2.5), the intermediate frames are encoded recursively as Bidirectional (B)-frames.

HEVC processes all type of frames in a block-wise manner. To adapt the encoding to the content, the frames are divided recursively into multiple blocks of pixels. The HEVC standard introduced a quad-tree structure for the block partitioning. Each upper block in the tree structure thus has four block children of the same size. Thus, the HEVC standard defines four different types of blocks in the quad-tree structure.

- Coding Tree Unit (CTU): is the largest block structure in HEVC. When building the quad-tree, the frame is first divided into CTUs of fixed size of 64x64 pixels

for instance.

- Coding Unit (CU): the previously obtained CTUs can be divided into 4 CUs. Each CU can also be divided recursively into 4 smaller CUs. Up to three levels of recursion are allowed in the HEVC standard, from 64x64 pixels down to 8x8 pixels. The choice of the prediction mode, intra or inter prediction (explained in the following subsections), is performed at the CU level. CUs within the CTU are processed in a Z-scanning order, or zig-zag order: from the top left CUs to the bottom right ones, going right to left.

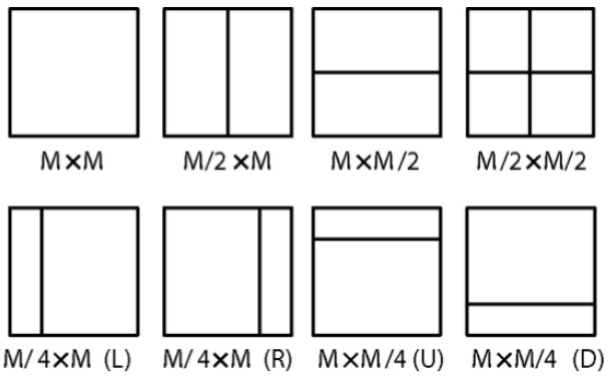


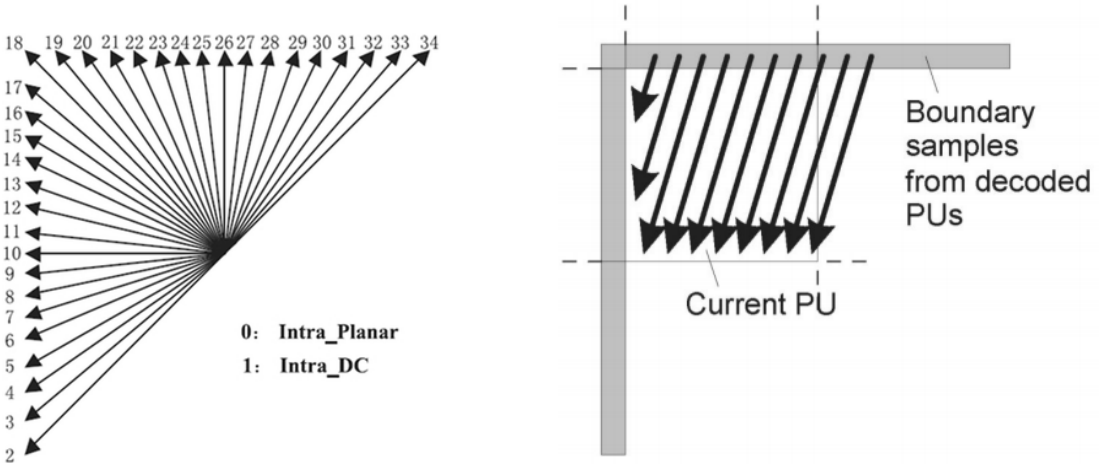

Figure 2.3: The eight possible PU partition schemes.

- Prediction Unit (PU): each CU can be divided into multiple PUs. The prediction information, motion vector for inter or mode index for intra, are estimated and stored at the PU level. A CU can contain up to four PUs. Several partitioning schemes are available and differ from the previous quad-tree partitioning. For the intra mode, only squared PUs are available, so an intra CU may only have one or four PUs. For the inter mode, eight configurations are defined as rectangular PUs are allowed: two squared PUs, three vertical rectangular PUs, and three horizontal rectangular PUs, as shown in Figure 2.3.

- Transform Unit (TU): each CU is also recursively divided into one or several TU. The transform and quantization steps are performed on the TU level. Each TU can be split into multiple smaller TUs in a quad-tree structure. TU sizes ranges from 32x32 to 4x4, and are also processed in a zigzag order within the CU.

  HEVC brings an interesting bitrate reduction mainly due to the prediction and transform of the residual error coding. In the following, the intra and inter prediction are explained.

- Intra prediction: the intra prediction mode is designed to exploit the spacial redundancy within the current frame. The intra prediction relies on the neighbouring reconstructed blocks pixels. For instance the I frame is only encoded using the




(a) (b)

Figure 2.4: (a) HEVC intra prediction modes and (b) intra prediction pixel samples available from the neighbouring reconstructed blocks.

intra prediction mode. Since the intra prediction does not use the temporal redundancy, the I frames have a high coding cost and represent a significant part of the total bit-rate of an encoded sequence. These reconstructed I frame were previously encoded and decoded, so their pixel values will also be available during the decoding process. As the blocks are processed in a zigzag scan, pixels on top and left of the current block can be used. As multiple block sizes are possible with HEVC, pixels on the bottom left and top right may also be retrieved for prediction in some cases. When processing the first blocks of the frame, no neighbor is available for the prediction. A padding operation is thus performed beforehand. Several methods, or modes, can be used to predict the current block values from the neighbouring pixels. The HEVC standard defines 35 intra prediction modes: DC, planar and 33 directional modes as illustrated in Figure 2.4.a. The DC prediction is defined as the average of the neighbouring pixels. The planar mode consists in a multi-directional prediction, horizontal and vertical, from the neighbouring pixels. The directional modes are used to predict the current pixel values by extending the neighbouring pixels in a given direction. The index of the mode is chosen and transmitted at the CU level. Along with the quantized residual error between the original block value and the reconstructed predicted value. The neighbouring pixels border used for a directional mode is depicted in Figure 2.4b. The chroma components (Cb and Cr) can only be predicted from five modes: planar, DC, horizontal, vertical and Direct Mode (DM). Prediction with the Direct Mode is performed by using the same mode selected from the Luma component. This mode relies on the strong correlation between the luma and chroma components [Beg18].

- Inter prediction: Aas mentioned before, the inter prediction is designed to leverage the temporal redundancy between consecutive frames. The basic idea is to use previously encoded and decoded frames as references to encode the current frame. Multiple reference frames can be used to encode the current frame.

  A P frame can be coded using intra prediction or inter prediction from past reference frames, while a B frame can be coded using both past and future reference frames, and intra prediction. Compared to the I and P frames, B frames have a significantly lower bit-rate. To be able to use future frames for prediction the encoding order differs from the temporal order. B predictive frames were introduced in H.264/AVC [WSBL03]. The inter prediction is performed by finding translational motion vectors for each prediction unit. The motion vectors are defined with a quarter pixel accuracy to obtain better prediction. The reference frame index and the motion vector parameters $(dx, dy)$ are encoded in the bitstream. The decoder will, then, be able to perform motion compensation and prediction. In order to reduce the size required to encode the motion vector parameters, a prediction is also performed. A motion vector prediction is obtained from the previously neighbouring PUs encoded with inter-prediction, or from motion vectors from reference frames. Then, only the difference (residual) between the predicted motion vector and the estimated one is actually stored in the bit-stream.

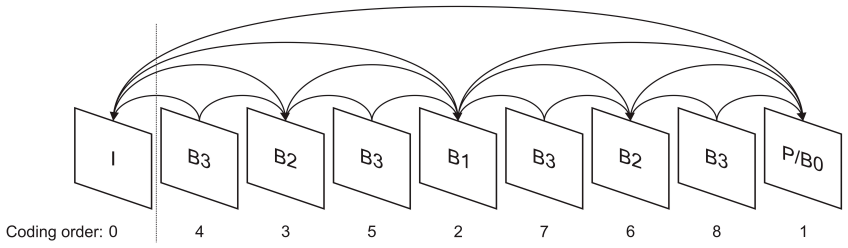


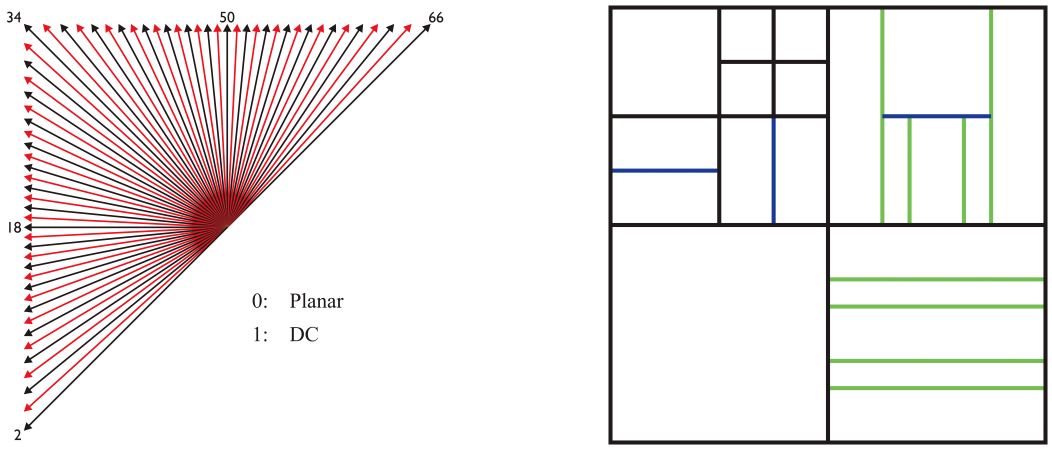

Figure 2.5: A traditional hierarchical GOP structure. P and B frames can be predicted from multiple reconstructed reference frames.

The HEVC standard defines two variants of inter prediction: "merge" and "skip". For these two modes, only the motion vector prediction is performed, there is no motion compensation step.

Up to five motion vector candidates are collected from neighbouring PUs, only the index of the selected one is stored in the bit-stream. Compared to the merge mode, the skip mode does not encode the block residual values. The reconstructed block is the same as the predicted one. Both these modes require less side information to transmit to the decoder and are computationally less expensive than classical inter prediction as they do not require the motion estimation. However, they rely on high temporal correlations as they are less accurate that the inter mode.

Transform: HEVC uses the classical DCT-II transform of TU of sizes from 32×32 to 4×4. The DST-VII is also used for specific case of Intra coded blocks of size 4×4.

Entropy coding: quantized residual errors and side information (i.e: frame indices, motion vectors ($d_x$, $d_y$), inter mode index) are coded using entropy coding. Context adaptive binary arithmetic coding CABAC is used for entropy coding. This is similar to the CABAC scheme in H.264/MPEG-4 AVC, but has undergone several improvements to mend its throughput speed (especially for parallel-processing architectures), its compression performance, and to reduce its context memory requirements [SOHW12].




(a) (b)

Figure 2.6: (a) Intra prediction modes in VVC and (b) block partitioning binary split and ternary split in VVC.

### 2.3.3 Versatile Video Coding

Based on HEVC, the Joint Video Exploration Team are currently developing a new video coding standard called Versatile Video Coding VVC [MWS17]. This latter reduces the bitrate compared to HEVC by almost 40-50% at the same visual quality [SHD+]. VVC outperforms HEVC by improving the coding tools such Intra/Inter predictions, block partitioning, transform module and loop filtering [RHPD19].

For intra prediction, VVC with 65 directional modes (with only 33 in HEVC) can have more detailed prediction and more precise prediction as shown in Figure 2.6a.

For Inter prediction, VVC uses advanced motion vector prediction, affine models and sub-block partitioning. Whereas in HEVC only square blocks were predicted, rectangular shapes are also possible in VVC. In addition to the binary block partitionning, VVC introduces the ternary split block partitionning, as shown in the Figure 2.6b. There are now multiple splits which are embedded in a multiple tree structure.

## 2.4 Machine Learning and Deep Learning

### 2.4.1 General Introduction

Humans and animals have the lifelong ability to learn, acquire, control and develop their knowledge and skills. This ability, referred to as lifelong learning, is mediated by a rich set of neural cognitive mechanisms that together contribute to the development and allocation of sensory skills, cognition and learning. It ,furthermore, allows living creatures to identify objects and understand accidents as well as to enhance and restore memory in the long run.

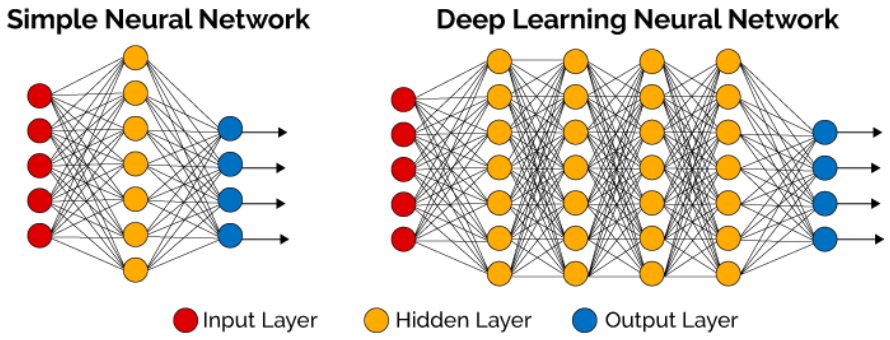



Figure 2.7: Neural network design.

Machine learning technology is attracting scientists from every domain. It consists in imitating the human intelligence and in particular their biological neural network, going from a simple network with 2 layers used for classification, to a deep network with multiple layers for image processing [LCCL08]. Deep learning belongs to machine learning technology and has the particularity of having computer models, called deep artificial neural networks or deep networks. It is composed of several processing layers (generally more than three).

In the latest studies, deep networks showed a great performance in image and video compression [CHB17, LYT+17, PMG+17]. Therefore, the next subsections define the so called Neural Networks and explain the architecture of a Convolutional Neural Network.

### 2.4.2 Neural Networks

Similar to a biological neural network, a simple Neural Network (NN) is defined by a set of interconnected nodes characterized by weights and biases and a linear activation function, distributed on 2 layers: one Hidden layer and the Output layer, as shown in Figure 2.7.

$$(\mathrm{z}) = sigmoid(w_1 x + b_1) \tag{2.5}$$

$$(\mathrm{y}) = sigmoid(w_2 z + b_2) \tag{2.6}$$

where $x$ is the input, $z$ the output of the first layer and $y$ is the output or predicted value, $w_1$, $w_2$ are weights, $b_1$ and $b_2$ are biases, sigmoid is a simple example of an activation function as shown in Figure 2.8.

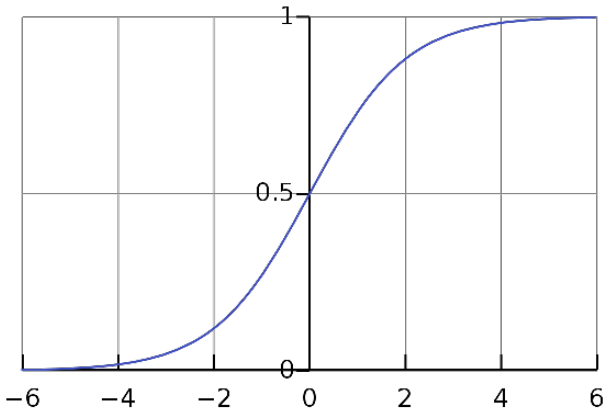



Figure 2.8: Sigmoid function.

Computing the output $y$ is called feedforward, where initially the weights and biases are set randomly. The goal of training the neural network is to update these parameters in order to obtain an output as close as possible to the real desired output [EPdRH02]. This phase is called backpropagation. However, to measure the performance of the NN, one needs a loss function to evaluate how far the predicted output is from the real one. Many loss functions are possible, the easiest and simplest one is the Sum of square errors. Finally, the training of a neural network stops when the error between predicted $y$ and the real output is lower than a certain threshold. After that, the NN is ready for testing with real life data [Pom91, Jia99].

### 2.4.3 Convolutional Neural Network

For image processing, NN can be used efficiently. However, linear functions are replaced with non-linear or convolutional functions.

A simple CNN is a sequence of layers, and every layer of a CNN transforms one volume of activations to another through a differentiable function. Three main types of layers are used to build CNN architectures: Convolutional Layer, Pooling Layer, and Fully-Connected Layer (Figure 2.9).

The feature map is the output of one filter applied to the previous layer. A given filter is drawn across the entire previous layer, moved one pixel at a time. Each position results in an activation of the neuron and the output is collected in the feature map.

**The convolutional layer** is the core building block of a CNN. The layer's parameters consist of a set of learnable filters (or kernels), which have a small receptive field, but extend through the full depth of the input volume. During the forward pass, each filter

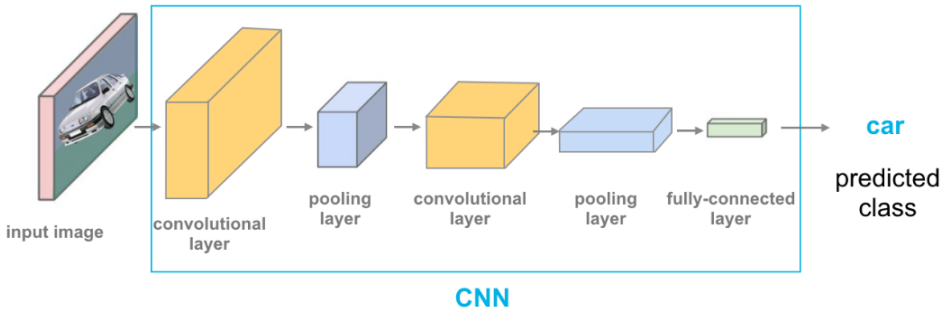
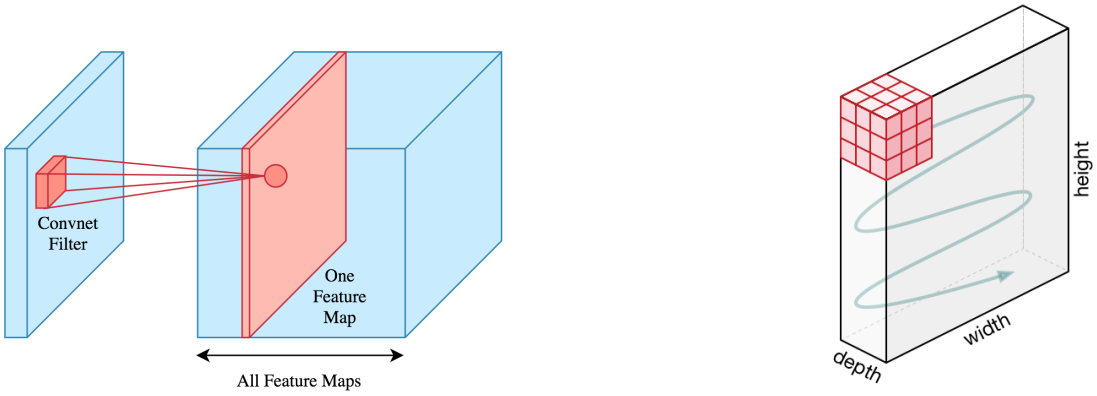



Figure 2.9: Example of CNN that uses some layers, looks at an image and outputs the correct class for it.

is convolved across the width and height of the input volume. It computes the dot product between the entries of the filter and the inputs and produces a 2-dimensional activation map of that filter. As a result, the network learns filters that are activated when it detects some specific type of feature at some spatial position in the inputs.




(a) (b)

Figure 2.10: (a) Convolutional Neural Layer: a matrix known as a kernel is passed over the input matrix to create a feature map for the next layer and (b)a CNN arranges its neurons in three dimensions (width, height, depth), as visualized in one of the layers and the mouvement of filter on the input

**Pooling layer** is common to periodically insert a Pooling layer in-between successive Convolutional layers in a CNN architecture. Its function is to progressively reduce the spatial size(down-scale) of the representation in order to reduce the amount of parameters and computation in the network. Pooling layer operates on each feature map independently. The most common approach used in pooling is max pooling (by taking the maximum value from the sub-array from the input array).

**Rectified linear units Layer,** Relu in neural networks is the max function(x,0) with input x e.g. matrix from a convolved image. ReLU then sets all negative values in the matrix x to zero and all other values are kept constant.

**Fully connected layer** takes an input volume (whatever the output is of the conv or ReLU or pool layer preceding it) and outputs an $N$ dimensional vector where $N$ is the number of classes that the system has to choose from.

### 2.4.4 Generative Adversarial Network

Recent researched proves its success in a variety of applications, such super-resolution, image recognition and object detection. Therefore, one can use it to predict images and compare its performance with CNN in the same environment configurations and dataset.

A GAN is an artificial intelligence technique for creating perfect imitations of images or other data. A GAN is a recent machine learning technique. It is based on the competition between two networks within a single framework.

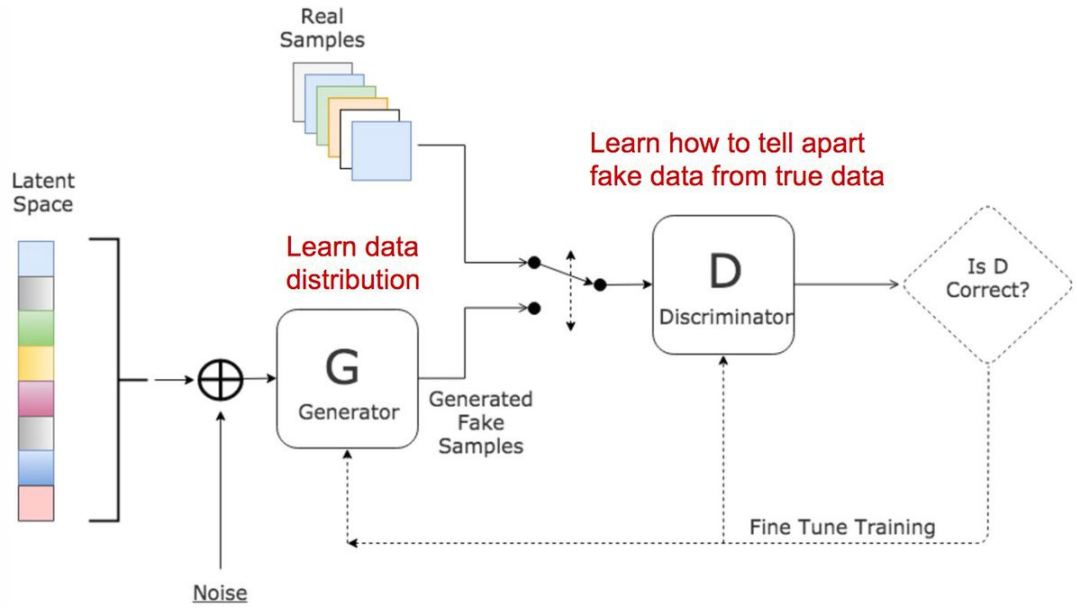



Figure 2.11: GAN Architecture.

These two networks are called "generator" and "discriminator". The generator is a type of convolutional neural network whose role is to create new instances of an object. The discriminator, on the other hand, is a "deconvolutional" neural network that determines the authenticity of the object or whether or not it is part of a data set. During the training process, these two entities are in competition and this is what allows them to improve their respective behaviours. This is called retropropagation.

In details, GAN takes an random input and tries to generate a sample of data. In the figure 2.11, we can see that generator $G(z)$ takes a sample input $z$ following a probability distribution $p(z)$. It then generates a data which is fed into a discriminator network $D(x)$. The task of Discriminator Network is to take input either from the real data or from the generator and try to predict whether the input is real or generated. It takes an input $x$ from $P_{data}(x)$ where $P_{data}(x)$ is our real data distribution. $D(x)$ then solves a binary classification problem using sigmoid function giving output in the range 0 to 1. In other words, D and G play the following two-player minimax game with value function $V(G, D)$:

$$\begin{aligned} &\min_G \max_D V\ (G, D) \\ &V\ (G, D) = \mathbb{E}_{x \backsim P_{data}}[\log D(x)] + \mathbb{E}_{z \backsim P_z}[1 - \log D(G(z))] \end{aligned} \tag{2.7}$$

In our function $V(D, G)$, the first term is entropy that the data from real distribution $(P_{data}(x))$ passes through the discriminator . The discriminator tries to maximize this to 1. The second term is entropy that the data from random input $(p(z))$ passes through the generator, which then generates a fake sample which is then passed through the discriminator to identify the fakeness [Ga14].

On the other hand, the task of generator is exactly the opposite, i.e. it tries to minimize the function $V$ so that the differentiation between real and fake data is bare minimum.

## 2.5 Existing Light Field Image Compression Techniques

### 2.5.1 Introduction

The Light Field compression can be classified into 2 categories: The lossy and the lossless techniques. Many studies have investigated lossy and lossless compression of LF imaging leveraging both spatial and angular redundancies in the image using different types of representations as illustrated in Figure 2.12. This part discusses in details the state-of-the-art of the different existing LF image compression techniques. Section 2.5.2 presents the lossless LF image coding. Then, in Section 2.5.3, the second category is presented.

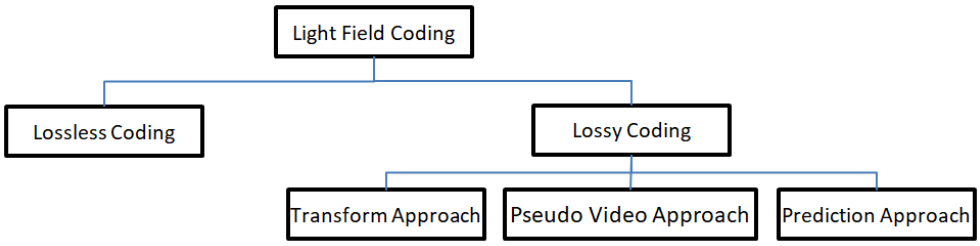



Figure 2.12: Classification of LF image coding techniques.

### 2.5.2 Lossless Coding Light Field

For scientific and cinema production, high quality images are needed. Lossless LF coding is then required. The fundamental approach for lossless LF coding is to predict the macroblocks and code the residual prediction error. The basic lossless LF coding scheme consists of using the HEVC reference Model (HM) or the AVC model, where LF image is fed as a pseudo video sequence with a spiral order scan [HART17]. In Table 2.1, the reported file sizes are directory sizes containing all necessary files for decoding.

Schiopu et al. [SM18] propose a macro-pixel prediction method based on CNN. They predict each macro-pixel based on a volume of six macro-pixels generated from its immediate causal neighborhood. Then, the resulting macro-pixel residuals are encoded by the reference CABAC (Context-based, adaptive, lossless image coding).

Table 2.1: Compressed file sizes in mega bytes [HART17]. The size of the original file is 183 MB.

| LF Image | HEVC | AVC | HM |
|---|---|---|---|
| Bikes | 82.0 | 80.8 | 88.32 |
| Danger_de_Mort | 87.0 | 86.5 | 95.46 |
| Color_Chart_1 | 83.3 | 81.9 | 96.40 |
| ISO_Chart_12 | 78.3 | 78.5 | 84.96 |

Gabbouj et al. in [SGGH17] proposed a lossless compression predictive method based on context modeling that exploits the redundancy of sub-aperture views. For each intermediate view a one neighboring reference view is selected and segmented. The residuals errors divided into two sets small and big compared to a threshold, are encoded by entropy coding.

Perra et al. [Per15] propose a lossless compression scheme based on adaptive prediction. The micro images composing a plenoptic image are processed by an adaptive prediction tool, aiming at reducing data correlation before entropy coding takes place.

### 2.5.3 Lossy Coding Light Field

While lossless compression rebuilds the exact data, lossy compression removes an unnecessary and undetectable part of the data, which is undetectable. These techniques includes the three following approaches:

Transform coding approaches: [JPFG17, Agg11, DQW04] This approach consists in transforming the LF image from its raw format to another basis which is more suitable for compression. Xiang et al. [JPFG17] actually aims at reducing the dimension of the captured data via a low rank approximation of views aligned by homographies which are jointly optimized with the low rank model, considering both a single homography per view and per depth plane.

Xu et al. [DQW04] proposed a wavelet packet-based Light Field compression method. Firstly, the original light images are decomposed into subbands. The latter are divided into two parts: one contains the subbands which have significant coefficients, large rela-

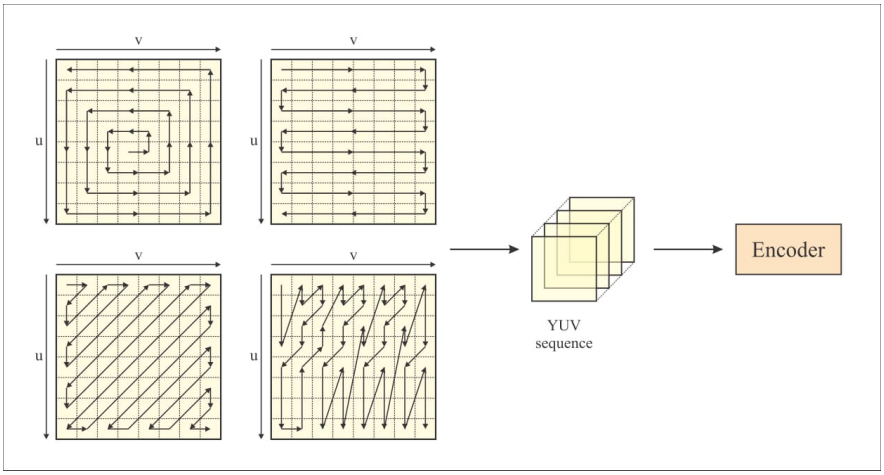



Figure 2.13: Pseudo video sequence of Light Field image with four scan orders.

tive energies and large correlations while the other contains the subbands which mainly provide isolated information of each image. Each subband is coded independently. In 2014, the JPEG standardization committee launched a new activity called JPEG Pleno [jpe18]. It aims to provide a standard framework for efficient storage and transmission of new imaging modalities (such as point-cloud, holographic and Light Field), which, when necessary, can also offer interoperability with existing standards, such as legacy JPEG and JPEG 2000 formats. Since then, JPEG committee has been actively pursuing the definition of a new standard representation and compression algorithm for LF images [EFPS16].

The pseudo sequence coding approach: This approach consists first in rearranging Light Field elements (usually sub-aperture images) in a specific order to produce a pseudo-video sequence [ZCYH16], which is then encoded with a classical 2D hybrid (intra and inter predictions) video encoder [LSOJ14]. This approach might, also, employ Multi-View extension of High Efficiency Video Coding (MV-HEVC) [AOS17].

Waqas et al. [AOS17] proposed a compression scheme based on MV-HEVC. It interprets each row of subaperture views as frames of a multi-view sequence that are compressed by using MV-HEVC. Inevitably, this method invests similarity between the multi-view sequences as well. Liu et al. [LWL$^+$16] proposed a compression of LF images based on pseudo-sequences of sub-aperture images. Firstly, the lenslet image is converted from YUV420 to RGB444 color space. Then, the lenslet is processed to obtain the multiple views that compose the Light Field data structure. The views are color and gamma corrected to be converted back to YUV420. A subset of them is then rearranged in a specific coding order that accounts for similarities between adjacent views. It is coded using the JEM encoder illustrated in Figure 2.14. Li et al. [LSOJ14] incorporated a full inter prediction scheme in HEVC intra prediction mode that explicitly embodied the redundancy in lenslet images. Perra et al. [PA16] proposed a method that partitions the raw Light Field into tiles of equal size. Then these tiles are ordered

as a pseudo-temporal sequence in order to adapt the data. Later on, they are rearranged in a pseudo sequence video to subsequent HEVC temporal predictive coding.

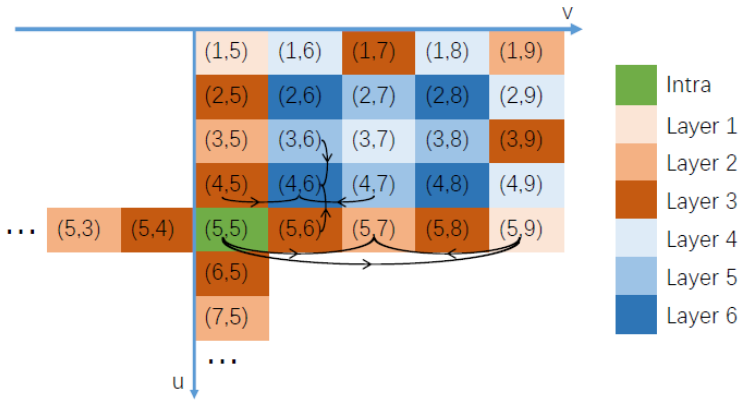



Figure 2.14: The coding order and prediction structure taking 9×9 views as an example for illustration. Only a portion of views is shown, with color indicating its layer. The arrows show prediction relations, each from reference to target view [LWL+16].

The predictive-coding approach:This approach takes advantage of the intrinsic high redundancy of LF images. In particular, instead of encoding all LF sub-views, only sparsely sampled LF sub-views are encoded and the remaining sub-views are reconstructed from the coded sub-views at the decoder side. This approach gained the attention of a lot of researchers.

In [ZC17], Shengyang et al. proposed a powerful LF coding scheme. The distance between adjacent cameras is a constant scalar. Mathematically, the LF image is modelled by a 4D function

$$L : \Omega \times \Pi\{\Longrightarrow \mathbb{R}\}, (\{\rho, \varphi\}) = \mathrm{L}(\{\rho, \varphi\}), \{\rho \in \Omega\} \tag{2.8}$$

where $\rho$ is a scene point, $\Omega$ represents the image plane and $\varphi = (\mathrm{u,v})^T$ denotes the offset of one view w.r.t. the center view in lens plane. As shown in Figure 2.15, this scheme consists in coding a sparse set of LF views ($\mathrm{S_A}$) using HEVC and then linearly approximating the other views ($\mathrm{S_B}$) and sending only the approximation coefficients to the decoder after quantization and entropy coding. The LA prior of the dropped vectorized view $j$ ($\mathrm{V}_j$) is given as follows:

$$V_j \approx \frac{1}{\Sigma x_m} \sum_{m \neq j}^{M} x_m V_m, \quad 2 \leq M \leq N \tag{2.9}$$

where $M$ is the number of selected reference views and $N$ is the total view number, $1 \leq \mathrm{m} \leq \mathrm{M}$ and $x_m$ are the weight coefficients. This coding scheme enables between 37.41% and 45.51% Bjøntegaard Delta Bit Rate (BD-BR) reduction on average compared to the HEVC encoding all sub-aperture views (HM-All) applied on a selected set of LF images

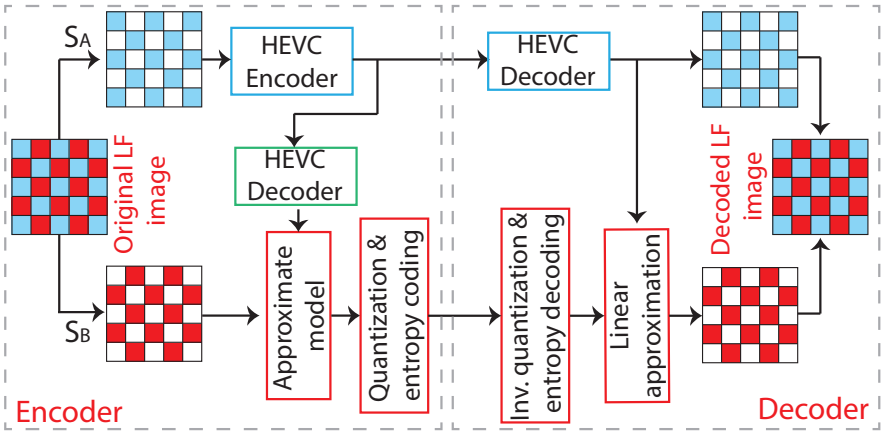



Figure 2.15: Linear approximation coding scheme [SZ17].

from EPFL dataset [RE16]. This gain is achieved when half of views are encoded to be transmitted to the decoder and other half of views are linearly approximated.

In recent years, supervised learning with CNN has witnessed huge adoption in computer vision applications like super resolution. In the predictive coding approach, different LF images are predicted by exploiting the redundancy with neighboring views using a CNN block.

In [KWR16], authors proposed a learning-based approach to synthesize new views from a sparse set of input views. The LF synthesis scheme is composed of disparity and color estimation components (Figure 2.16). Authors use two sequential CNNs to model these two components and train both networks simultaneously by minimizing the error between the synthesized and ground truth images. They used only four corner sub-aperture views from the LF captured by the Lytro Illum camera to synthesize high-quality images that are superior to the state-of-the-art techniques. As shown in Figure 2.16, a set of features (mean and standard deviation) of a sparse set of views are fed to the first CNN that estimates the disparity at an intermediate view using Equation 2.10.

$$D_q = g_d(K), \tag{2.10}$$

This equation models how the estimated disparity $D_q$ at the novel view at position $q$ is generated from the set of $K$ features including the mean and standard deviation. Finally, the second CNN generates the final intermediate view using Equation 2.11.

$$F_q = g_c(H), \tag{2.11}$$

where $F_q$ represents the image at the intermediate view, $H$ the feature set and $g_c$ defines the relationship between these features and the final intermediate image.

Likewise, Gupta et al. [GJK+17] combined their results to recover a high resolution 4D LF from a single coded 2D image with two branches network architecture a traditional autoencoder and 4D convolution layers. Jiang et al. [JLG17] proposed a Light Field compression scheme using depth image-based rendering approach. A sparse set

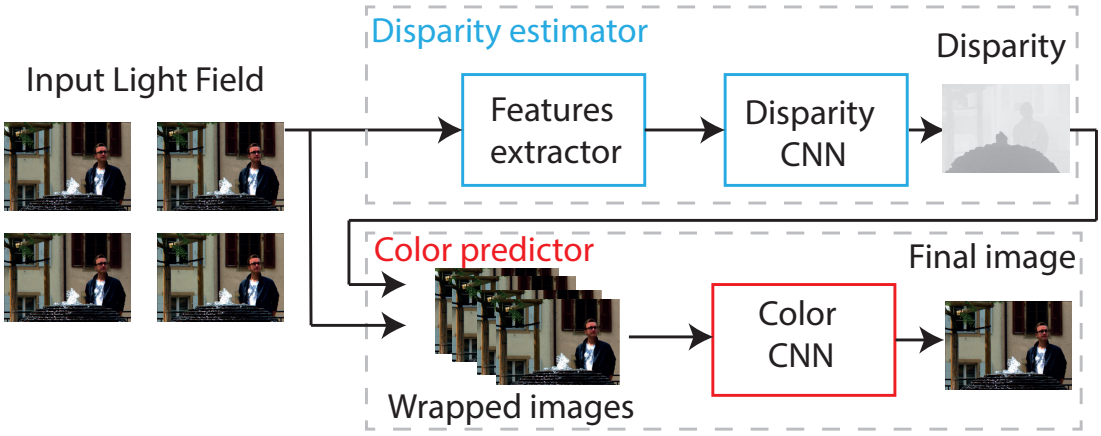



Figure 2.16: Deep learning views synthesis [KWR16].

of views is selected and encoded with the standard HM and transmitted to the decoder side. The depth image and low rank matrix completion are used in three main blocks to synthesize the entire LF image from the decoded views. First block to estimated a warped disparity, then the second to synthesis the warped color image and the last block to synthesis the final color image.

Wu et al. [WZW+17] use the epipolar plane image (EPI) representation of LF to reconstruct the whole LF image by CNN-based angular detail restoration on EPI. To avoid the ghosting effects caused by the information asymmetry, the spatial low frequency information of the EPI is extracted via EPI blur and used as input to the network to recover the angular detail. The non-blind deblur operation is used to restore the spatial detail that suppressed by the EPI blur.

Dib et al. [DPG19] proposed a LF compression scheme based on the Fourier Disparity Layer representation. The LF is divided into several subsets of views. The first subset is encoded with a standard 2D video encoder HEVC, while the second subset of views is predicted by the Fourier Disparity Layer view synthesis. For better reconstruction, the residue data of synthesized uncoded views are compressed and transmitted to the decoder side.

Due to the non-linearity in LF images caused by the angular displacement, Zhao et al. [ZWJ+18] applied a non-linear deep-learning-based view synthesis network to boost the performance of the LF images compression. In particular, they use 2 CNNs. The first CNN with 6 layers is used eventually to synthesize the missing LF sub-views. It takes as inputs all the accessible decoded sub-views in clockwise order from the current viewpoint. The second CNN is used to enhance the quality of the reconstructed LF sub-views. The loss function used to measure the difference between the enhanced sub-views and the dropped viewpoints is $\ell 2$-norm. The method proposed by Zhao et al. shows a high efficiency of their deep learning based scheme.

Jia et al. [JZW+18] proposed a similar LF image compression. The main difference is that they use a GAN (with a generator and discriminator) instead of a single CNN. For quality refinement, Jia et al. encoded and transmitted the residual errors of synthesized

uncoded views.

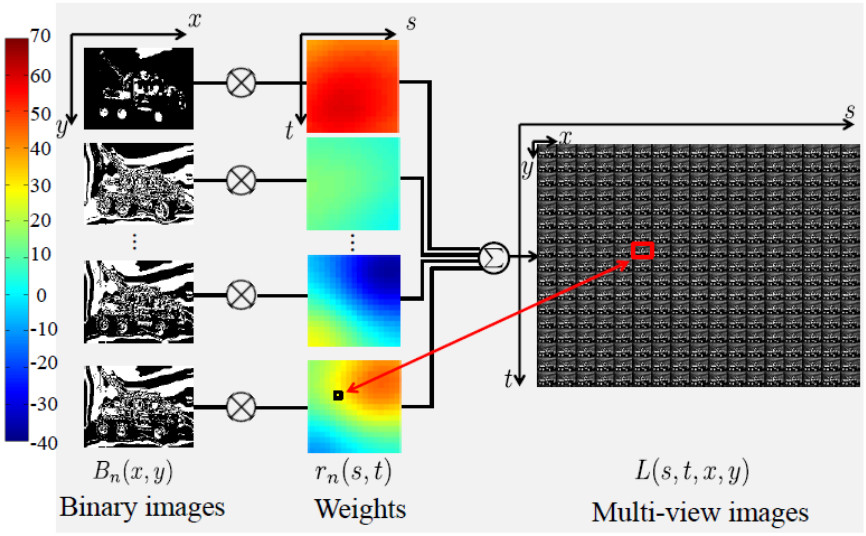



Figure 2.17: Light Field representation using binary images and weights [KTF18].

Huang et al. [HAS+18] synthesized the missing views using depth-based image rendering technique, where depth maps (D) are generated from the epipolar images using the equation 2.12

$$D = f/(1 - tan(\alpha)), \tag{2.12}$$

where $f$ and $\alpha_i$ are camera focal length and the slope in the Epipolar Plane Image (EPI), respectively.

Dib et al. [DLJG19] proposed a compression scheme for Light Field using super-ray based local low rank models. A novel method for disparity estimation and compensation was proposed so that the super-rays are constructed to yield the lowest approximation error for a given rank. This representation is based on two low rank models, one for the central view pixels that are visible in all views and while the other is for occlusions.

Komatsu et al. [KTF18] proposed a more simple coding scheme. They modeled the LF image as a set of binary images $B_n(x, y)$ combined with a set of weights $r_n(s, t)$ where the viewpoints, which are arranged in a 2-D grid, are specified as $(s, t)$ and the pixels are indicated as $(x, y)$- one set of weights for each RGB component- and shown in Figure 2.17. These weights are computed by minimizing the mean square error between the original image and the reconstructed one as shown in equation 2.13

$$\arg \min_{B_n(x,y), r_n(s,t)} \sum_{s,t,x,y} |L(x, y, s, t) - \sum_{n=1}^{N} B_n(x, y) \times r_n(s, t)|^2, \tag{2.13}$$

where $L(s, t, x, y)$ is the original Light Field image, $N$ represents the number of binary images and $n = 1,...N$. Only these information are sent to the decoder, this offers

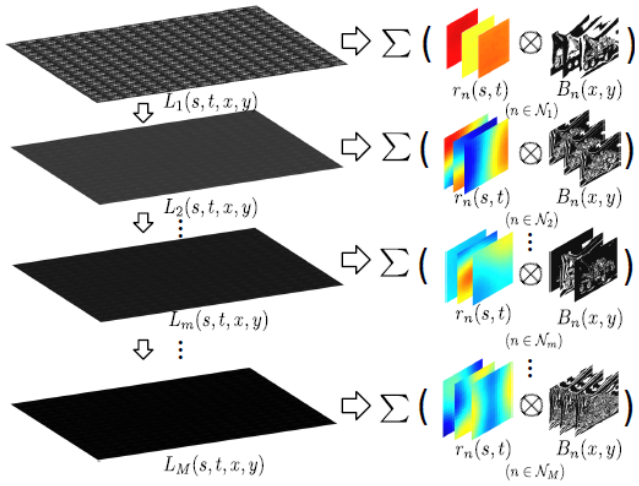



Figure 2.18: Scalable LF compression by weighted binary images [KTF18].

a good compression ratio that is comparable with the existing methods. Komatsu extended their proposed method to a scalable scheme where the binary image can have different resolutions depending on the degree of granularity as shown in Figure 2.18.

One should note that in the predictive coding approach of LF image coding, a coding technique depends on the way the images are viewed. For instance, Amirpour et al. [APP18] proposed a new scan order which divides sub-aperture images into four regions and encodes them independently. Each quadrant uses the central sub-aperture as first reference and encodes the non-central sub-aperture images in a snake order.

Pinheiro et al. [APP+19] used the concept of macro images, where they group immediate neighoring images. Each view image along with its immediate neighboring view images which have higher similarity, are grouped and called a Macro View Image (MVI). Considering 15×15 view images decomposed from a raw lenslet image to result in a division into 25 MVIs, in addition to the different colors allocated to each MVI reveal the level of dependency, as shown in Figure 2.19.

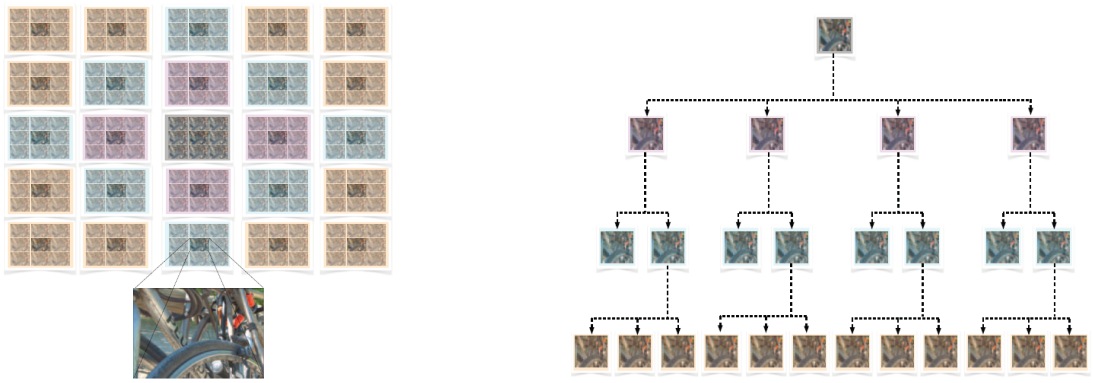
The third approach is broadly used in LF image compression as it proves a better performance and it is relatively more promising.

## 2.6 Light Field Visual Quality Evaluation

In order to compare compression performance of multiple coding techniques, one needs distortion and/or quality metrics.

### 2.6.1 Categorization of Objective Methods

Objective quality assessment methods, called by abuse of quality metric language, refers to metrics computed by mathematical tools in contrast to subjective evaluation. The



(a) (b)

Figure 2.19: (a) MVI groups of view images. (b) Dependency among centers of MVIs shown as a tree unit.

objective metrics can be classified into three categories depending on whether or not the reference video is available, as shown in Figure 2.20. According to ITU Recommendation J.143 [ITU], three video quality metrics have been defined: metrics with full reference (Full Reference, FR), metrics with reduced reference (Reduced Reference, RR) and metrics without reference (No Reference, NR) where Full-reference (FR) uses the full bandwidth video input. Reduced-reference (RR) uses lower bandwidth features extracted from the video input. As for, No-reference (NR), it has no information about the video input.

### 2.6.2 Objective Distortion Metrics

To evaluate the proposed algorithms in this thesis, four objective assessment tools were used: Mean Squared Error (MSE), Peak Signal to Noise Ratio (PSNR) and Structural Similarity (SSIM). These three metrics are briefly introduced as follows:

Mean squared error MSE is a mean squared difference between the original image $A$ and distorted image $B$. The mathematical definition for MSE is:

$$\text{MSE} = \frac{1}{M \times N} \sum_{i=1}^{M} \sum_{j=1}^{N} (A_{ij} - B_{ij})^2 \tag{2.14}$$

Where $\text{A}_{ij}$ and $\text{B}_{ij}$ are the pixel value at position $(i, j)$ in the original image and distorted image respectively.

Peak signal to noise ratio PSNR measures the distortion of a retrieved signal compared to its original version [HZ10]. The PSNR can be used to assess the fidelity between the original image A and distorted image B. The PSNR is computed pixel-wise:

$$\text{PSNR} = 20 \log_{10} \frac{255}{\sqrt{MSE}} \tag{2.15}$$

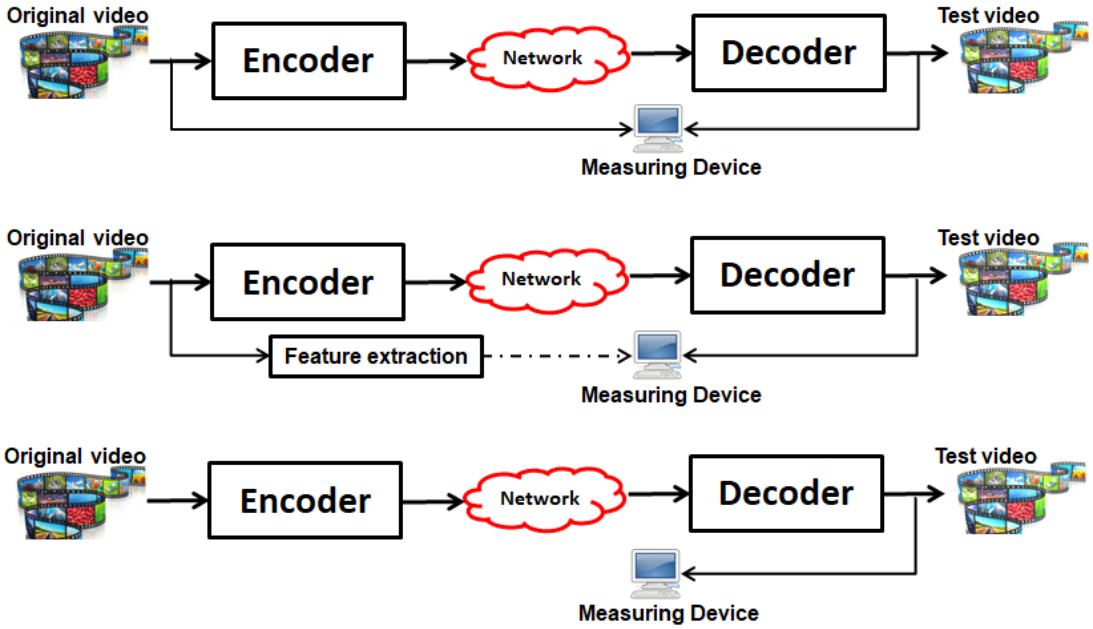



Figure 2.20: Three categories of video quality metrics

In particular, the weighted PSNR (WPSNR) is defined as:

$$PSNR_{YUV}(k,l) = \frac{(6 \times PSNR_Y(k,l) + PSNR_U(k,l) + PSNR_V(k,l))}{8} \tag{2.16}$$

where $K$, $L$ are the number of sub-aperture images in the whole LF for each line and column respectively and $k$ and $l$ are the indexes of the sub-aperture images.
The mean of sub-aperture images $\overline{PSNR}_{YUV}$ is subsequentially computed to have an average value for PSNR for Y channel and for YUV

$$\overline{PSNR}_{YUV} = \frac{1}{((K)(L))} \sum_{k=1}^{K} \sum_{l=1}^{L} (PSNR_{YUV}(k,l)) \tag{2.17}$$

SSIM is a method for measuring the similarity between two images. The SSIM index can be viewed as a quality measure of one of the images being compared, provided the other image is regarded as of perfect quality [ZBSS04].

$$SSIM(x,y) = l(x,y)s(x,y)c(x,y), \tag{2.18}$$

$$\mathrm{l(x,y)} = \frac{2\mu_x\mu_y + C_1}{\mu_x^2 + \mu_y^2 + C_1}, \tag{2.19}$$

$$\mathrm{s(x,y)} = \frac{\sigma_{xy} + C_3}{\sigma_x \times \sigma_y + C_3}, \tag{2.20}$$

$$\mathrm{c(x,y)} = \frac{2\sigma_x\sigma_y + C_2}{\sigma_x^2 + \sigma_y^2 + C_2}, \tag{2.21}$$

where $l$, $s$, $c$ are the luminance, structure and contrast similarity measurement comparison function respectively.

(a) (b) (c)

Figure 2.21: Visual quality of the image Fruit from the INRIA LF dataset [JPF+17], view (3,3) in the array sub-aperture 8 × 8. From left to right: original view, encoded respectively with HEVC (bitrate = 0.064 bits per pixel (bpp), PSNR = 31.5 dB, SSIM = 0.936) and VVC (bitrate = 0.073 bits per pixel (bpp), PSNR = 32.3 dB, SSIM = 0.947).

Bjøntegaard metric: When comparing two codec versions, differences can be measured on the distortion or bitrate level, either by observing the distortion improvements for a given bitrate, or measuring the bitrate reductions for a fixed distortion. Bjøntegaard et al. [Bjø01, hB08] introduced a simple framework to simplify the comparison between two prediction methods at multiple bit-rate levels. They proposed two metrics: the Bjøntegaard Delta Bit Rate (BD-BR) and the Bjøntegaard Delta PSNR (BD-PSNR), which respectively describe the average bit-rate or PSNR differences between two encoding methods. These metrics are computed for four points from a Rate Distortion curve (RD-curve).

The BD-PSNR measures the average PSNR difference between two RD-curves. The BD-PSNR is calculated using third degree polynomials over logarithmic bit-rates and PSNRs data points in equation. It is expressed in decibel (dB).

$$BD - PSNR = \frac{1}{r_H - r_L} \int_{r_l}^{r_h} (D_2(r) - D_1(r))dr \tag{2.22}$$

where the BD-BR is expressed in percentage. As it describes a bitrate difference, the BD-BR has negative values when there is an improvement, i.e. a bitrate reduction.

$$BD - Rate = \frac{1}{D_H - D_L} \int_{D_l}^{D_h} (r_2 - r_1)dr \tag{2.23}$$

with $r_h = \log(R_h)$, $r_l = \log(R_h)$ being the high and low boundary values of the output bit-range. $D_l$ and $D_2$ are the two RD-curves considered for comparison. Needless to mention, that we will be using the BR-rate, Bjøntegaard Delta SSIM (BD-SSIM) and BD-PSNR to evaluate the quality of the reconstructed images of our method with some of the state-of-the art techniques based on linear approximation [ZC17], deep learning [KWR16] and the standard 2D encoder [LLL+17].

## 2.7 Subjective Distortion Metrics

Subjective test is the most accurate way to measure the quality of a multimedia stream. More precisely, subjective evaluation stresses on visual aspects that are not considered in the objective tests. For instance, distortions on edges are not visually disturbing as distortions in the homogeneous zones. Such an example leads to a low objective metric, whether visually or subjectively it must give a high score.

In addition to the objective metrics, we subjectively evaluated our proposed method. The environment of the subjective experiment is equipped with recent tools that help viewers evaluate video and images. This environment was established according to the approved standard that is recommended in the ITU-R Rec. BT.1788. To be more specific, the quality and strength of lighting inside and paint colors used in addition to the adoption of the distances to assess the quality of images and videos for all compression methods.

### 2.7.1 Organization of Subjective Tests

In a quality subjective experience, organizers must meet a number of criteria in order to obtain reliable results. Therefore, the environment of the experiment and the test conditions must be strictly defined. Thus, the instructions given to observers, the stimuli present and the evaluation methodology are elements that can be fixed by the organizers. However, some factors related to the observers themselves, such as origin, culture or mood, can influence. The latter can be controlled by applying specific constraints such as vision tests and the use of several participants within the same experiment.

**The observers:** The subjective quality of visual content can vary considerably from one observer to another. Observers may be expert or non-expert depending on the objectives of the assessment. To reduce this variation gap, visual stimuli must be visualized by a set of observers. Recommendation ITU-R BT.500 [BT.12b] stipulates the use of at least 15 uninitiated or naive individuals to assess the quality of a visual stimulus. These observers must pass visual tests (Snellen scale) and have the ability to distinguish colours (Ishihara test, for example). Gender parity and the age of the participants are also important elements for a "quality" experience. Finally, the question of remuneration is raised.

**The test conditions:** Observers who have passed the visual tests are selected and the test conditions are explained to them : comparison methods, rating scales, etc. It is also recommended to start the experiment with a series of tests to familiarize users with the equipment of the experiment and to anchor their judgment.

A test session is typically composed of a set of potentially degraded stimuli (images or videos). The order of presentation of these stimuli must be random in order to avoid the observer's deconcentration and weariness. ITU also recommends that the duration of a test session should not exceed 30 minutes.

Table 2.2: General viewing conditions for subjective assessments in laboratory environment [BT.12b].

| Condition | Item | Values |
|---|---|---|
| *a* | Peak luminance on the screen ($cd/m^2$) | 150-250 |
| *b* | Ratio of luminance of inactive screen to peak luminance | <=0.02 |
| *c* | Ratio of the luminance of the screen when displaying only black level in a completely dark room, to that corresponding to peak white | approximately 0.01 |
| *d* | Maximum observation angle relative to the normal | 30° |
| *e* | Ratio of luminance of background behind picture monitor to peak luminance of picture | approximately 0.15 |
| *f* | Chromaticity of background | $D_{65}$ |
| *g* | Illumination from other sources | low |

For the laboratory viewing environment, ITU recommends many constraints to ensure the best viewing conditions. Thus, the lighting of the room, the display screen and the viewing distance must be respected.

Furthermore, the laboratory viewing environment is intended to provide critical conditions to check systems. General viewing conditions for subjective assessments in the laboratory environment Table 2.2 lists the general viewing conditions for subjective assessments in the laboratory environment on ITU-R Recommendation BT.500.

### 2.7.2 Subjective Evaluation Quality Assessment Methods

Although test conditions have a major impact on human judgment, the instructions given to observers also have a major influence on the production of subjective quality scores. As a result, various methods and protocols have been developed. The main task of a participant is to judge the quality of a degraded version of the video during its presentation. The way this version is presented to observers depends on whether or not the original version is present. Thus, three main families of methods standardized in ITU-R Recommendation ITU-R Rec. BT.500-10 have been proposed:

**Comparative methods:** Comparative methods Peer Comparison (PC) consist in simultaneously presenting observers with two versions of a video for which they are asked to quantify the existing qualitative relationship between these two versions. Thus, two comparative scales can be used: a discrete scale and a scale by category.

The category rating scale consists of a set of semantically defined indices where the participant must choose a particular category that represents his or her feelings. The discrete scale, on the other hand, offers more choice in scoring for participants in the experience. The subject is asked to compare the quality of the first video with that of the second within a time interval of less than 10 seconds, to judge which of these two videos is of better quality.

**Single stimulus methods:** A simple stimulus method, as its name suggests, is to show a video to an observer by asking him to judge its quality, without being accom-

panied by the original version. Two measures of this class of methods are generally used: the Single Stimulus Continuous Quality Scale (SSCQS) method and the Absolute Category Rating (ACR) method. In the literature, this last method is sometimes called Single Stimulus Impairment Scale (SSIS).

The main difference between these two methods is based on the use of the rating scale. For the ACR method, a discrete rating scale, generally 5 points, is used. Quality, in this scale, is often related to the perception of degradation. The SSCQS method uses a continuous scale, from 1 to 100, to guide the observer to the most appropriate score. The video is presented to the observer and then a grey image of less than 10 seconds (usually 5 seconds) is displayed on the screen during which the participant is asked to give it a quality score.

**Double stimulus methods:** This last category of subjective quality assessment methods consists of showing two stimuli to observers before rating their quality. As with single stimulus methods, two double stimulus measures are often used: The Double Stimuli Continuous Quality Scale (DSCQS) and the DCR (Degradation Category Rating) methods. The latter is sometimes called Double Stimulus Impairment Scale (DSIS). In the DSCQS method, a continuous scale is used while the DCR method uses a scale similar to that of the ACR (Absolute Category Rating) method. The difference between these two methods is not limited to the use of the rating scale but rather to the objective of the method itself. The objective of the DSCQS method is to evaluate a transmission system where the two versions presented to observers correspond to the input and output of this system. Participants do not know which version corresponds to the entry or exit. The DCR method simply measures the discomfort perceived by the observer when viewing the video. The observer is informed that the first version corresponds to the reference video while the second has the degraded video to which he must assign a quality score.

For all the methods described above, ITU recommended that the display time for an image or video should be around 10 seconds. However, this duration is relatively short and does not reflect a real situation, particularly for video stimuli. Thus, another double-stimulated method of continuous evaluation was proposed. This is the Simultaneous Double Stimulus for Continuous Evaluation (SDSCE) method where videos are presented side-by-side on the same screen or on two adjacent screens. Participants assign their quality score on a continuous basis. Due to its complexity, this method remains the least used in the literature. For our subjective evaluation, we used the double stimuli method.

### 2.7.3 International Telecommunications Union Recommendations

Assessing subjective quality involves psycho-visual tests where observers are asked to assess the quality of a video stimulus based on their own subjective judgement. The ITU, which is an international standard organization, has published a set of recommendations for the proper conduct of these subjective tests. The main ITU-T/R recommendations

related to the video quality evaluation method such as BT.500 (methodology for the subjective assessment of the quality of television pictures). The main ITU-T/R recommendations on video quality evaluation methods are listed in the Table below 2.3.

Table 2.3: The main recommendations of ITU for subjective quality assessment tests.

| ITU recommendation | Name |
|---|---|
| BT.500 | Recommendation ITU-R BT.500-13 (2012), methodology for the subjective assessment of the quality of television pictures. |
| P.910 | Recommendation ITU-T P.910 (2008), subjective video quality assessment methods for multimedia applications. |
| J.140 | Subjective picture quality assessment for digital cable television systems. |
| BT.1129 SDTV | Subjective assessment of standard definition digital television. |

## 2.8 Conclusion

This chapter briefly described the main 2D standard video techniques including HEVC and VVC. It, furthermore, displayed some advanced features such as image segmentation and showed that SLIC technology has the best efficiency. This literature review presented, also, the different approaches for the existing LF image compression techniques and revealed that the predictive approach is the most efficient one. The following chapter we will detail our proposed technique that is mainly based on linear approximation and deep learning. It will be compared to the state of the art techniques while using the different evaluation metrics as provided in this chapter.

# Chapter 3

# RDO-Based Light Field Image Coding Using Convolutional Neural Networks and Linear Approximation

## 3.1 Introduction

In this chapter, we propose an efficient LF image coding scheme with a rate distortion optimization (RDO) functionality. It is mainly based on a predictive approach using linear approximation and convolutional neural network. As shown in Figure 3.1, the encoder consists of coding a first sparse set of views with a standard encoder and a second sparse set with a linear approximation. While the last sparse set of views is input to a RDO block, where it will be either linearly approximated or simply dropped. Where at the decoder, we use a deep learning approach to synthesize the dropped views, followed with a post-processing pixel-matching-based scheme for a higher reconstruction quality.

Section 3.2 and 3.3 details the proposed method. Experimental results are presented in Section 3.4.

## 3.2 Hybrid 2D Video Codec and CNN Coding Scheme

Our basic scheme proposal referred in the following as Hybrid 2D video codec CNN (H2DC-CNN) coding scheme. The flowchart of the H2DC-CNN coding scheme is shown in Figure 3.2 and its main blocks are explained in the following. As shown in Figure 3.2, at the encoder side, for a given LF image $L$ constituted by sub-aperture views, a sparse set of reference views are selected ($S_R$: 8 corner and center views) and re-arranged in pseudo-video sequence. The latter is then compressed with a 2D video encoder standard. Next, a second sparse set of views ($S_E$: 7 adjacent views) are lin-

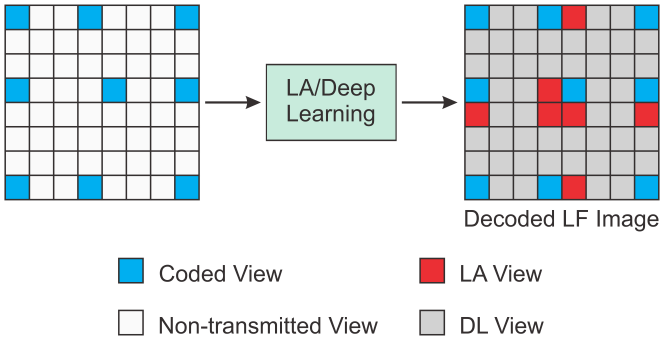



Figure 3.1: Global concept of our proposed scheme.

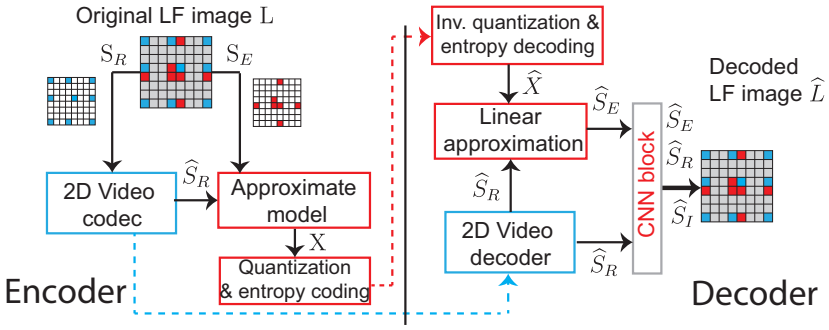



Figure 3.2: Block diagram of the proposed H2DC-CNN LF coding scheme.

early approximated [ZC17] with the decoded reference views ($\hat{S}_R$). For the views of $S_E$, only the coefficients of LA are transmitted to the decoder, thus allowing to reduce transmission bandwidth. The LA prior of the view $V_j$ is given as follows [ZC17]:

$$V_j \approx \frac{1}{\Sigma x_m} \sum_{m \neq j}^{M} x_m V_m, \qquad 2 \leq M \leq N \tag{3.1}$$

where $M$ is the number of selected reference views and $N$ is the total number of views, while $x_m$ are the weight coefficients of the vector $X$ with $1 \leq m \leq M$. The video bitstream and the weight coefficients ($X$) are both sent to the decoder.

At the decoder side, the reference views are first decoded ($\hat{S}_R$) and then jointly used with the weight coefficients ($\hat{X}$) to linearly approximate the $\hat{S}_E$ set. The two sets of views ($\hat{S}_R$ and $\hat{S}_E$) are then fed to the CNN block that synthesizes the remaining views (illustrated with gray color in the Figure 3.2). The CNN block includes two phases: a disparity estimator and color predictor, which are performed by two sequential CNNs [KWR16]. Based on the features extracted from the sparse input views, a four layer CNN estimates the disparity of the dropped views. The second CNN uses all the warped views, derived from the first CNN, along with a few other features to predict the color and synthesize the dropped views.

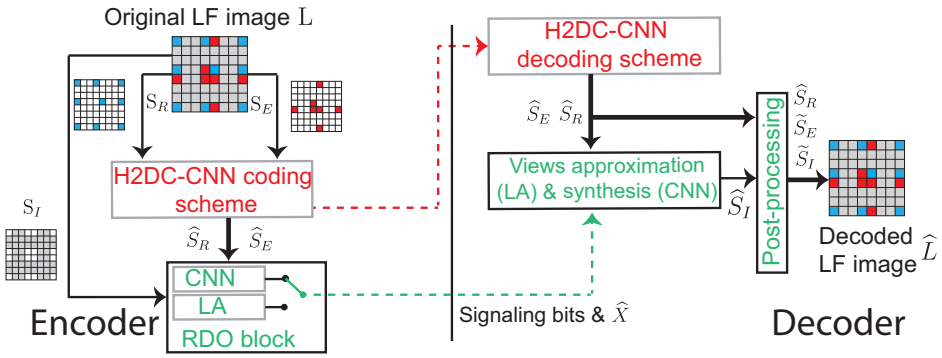



Figure 3.3: RDO-based Light Field coding using CNN and LA Scheme. Proposed LF image coding scheme (the new blocks are highlighted in green)

.

The obtained gain in coding efficiency with the H2DC-CNN scheme is about 30% compared to the state-of-the-art solutions [BHD+18]. However, while this method is very efficient at low bitrate, it is not so efficient for providing a high quality view at high bitrate compared to the pseudo-video sequence coding approach. Moreover, for some video sequences, we have noticed that the views linearly approximated have better quality than when they are synthesized by the CNN block. Therefore, we propose in the next section three main contributions to overcome aforementioned limitations and further increase the quality of the reconstructed views at both low and high bitrates.

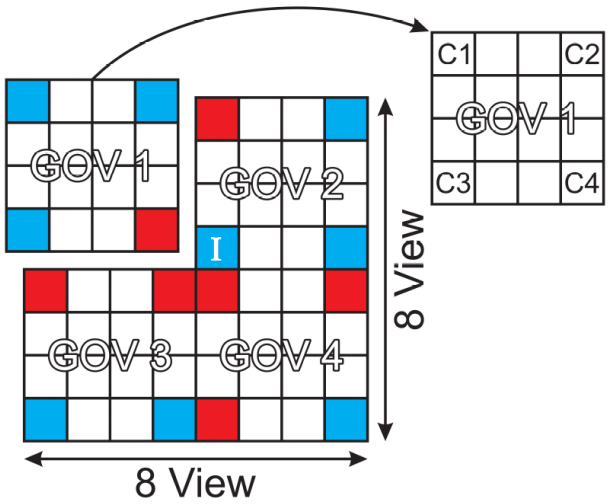
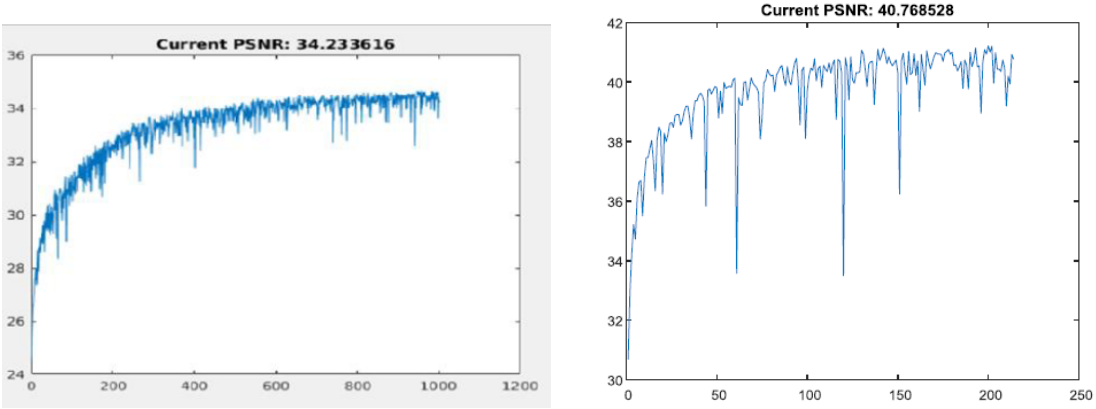



Figure 3.4: Sub-aperture representation of a LF image splitted into 4 groups of views (GOV). Each group of views takes the 4 corner views as reference, while view I represents the position of the intra frame.

## 3.3 Proposed Method

### 3.3.1 Configuration Settings

In the proposed scheme, we consider the sub-aperture based representation of LF image of $8 \times 8$ views. It consists in dividing the plenoptic image into four Groups of Views (GOV) ($4 \times 4$ views each) as illustrated in the Figure 3.4. We can notice that the performance of the training are more efficient with a GOVs as shown in the Figure 3.5.



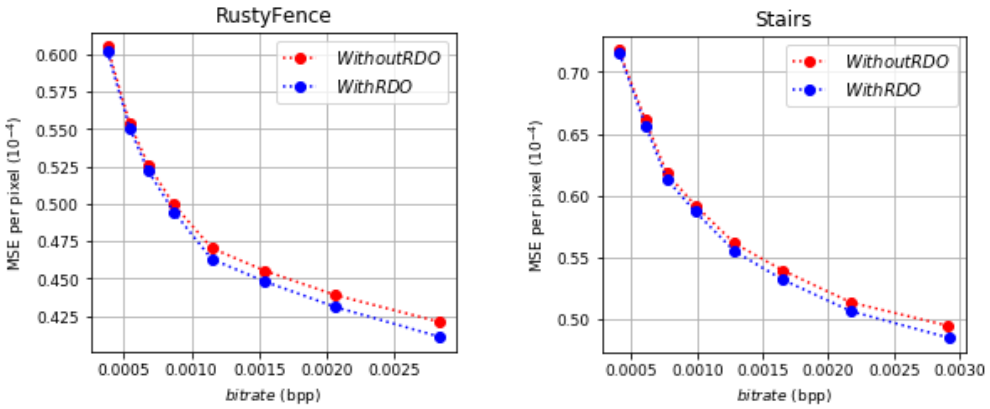

(a) Training 8×8 (b) Training 4×4

Figure 3.5: Quality performance for LF images 8 * 8 views: (a) with 4 views as references and (b) with four GOV, 16 views as references.

### 3.3.2 Global Framework

For each GOV, we take the 4 corners as reference in order to synthesize the novels views. In total, the number of references views is 16 for the whole LF image. As first step, we select a sparse set of sub-aperture views ($S_R$ in blue) with specific position that give the best result after testing all possible combinations. Then, we rearrange the nine $S_R$ views into a pseudo sequence (spiral order scan) and encode it with a simple JEM encoder with chrominance downscale, e.g yuv 420. In the second step, we estimate the 7 adjacent views set ($S_E$ in red) using linear approximation explained in Section 3.2. For each frame in the dropped views set $S_E$, we linearly approximate the views with the decoded views in $S_R$ set. An approximation model is used to optimize the reconstruction of the weight coefficients $X$, by using the Spectral Projected Gradient for L1 (SPGL1) functions. This one generates the coefficients for one target view at each time and for each channel color separately (i.e. rgb, 3 channels).
As this vector $X$ contains floating point values, we quantize $X$ at 16 bits before encoding it with entropy coding. The JEM bitstream encoding the $S_R$ set of views with the quantized and entropy coded linear coefficients are sent to the decoder.

In order to achieve bitrate reduction while maintaining high visual quality at different ranges of bitrate, we made three different improvements. Specifically, we introduced a RDO stage to make the right choice (LA vs. CNN) for the dropped intermediate views ($S_I$). In addition, we propose to efficiently tune the quality of the central view, since all the predicted, estimated and synthesized views are based on it. Finally, a post processing step is applied to the approximated and synthesized views to further enhance the visual quality of the decoded LF image. The block diagram of the proposed scheme is illustrated in Figure 3.3 and the newly added blocks are highlighted in green.




(a) (b)

Figure 3.6: MSE per pixel based comparison with and without RDO of *Rusty-Fence and Stairs* LF images.

### 3.3.3 Central View Quality Tuning

The Central View (CV) of the LF image, illustrated in Figure 3.4 at the position (4, 5), is coded as an Intra frame. It is used as a reference for the prediction of every other frames. In addition, it is exploited by the LA and CNN blocks to generate the dropped views. Thus, the quality of this CV is a key factor for the prediction and generation of other views. Therefore, we must be careful in fixing the quality of the CV.

A simple and efficient way to provide a CV with a high quality is to assign it a QP value lower than the global one used for the rest of the views: $Q_{intra} = Q + Q_{offset}$, where $Q_{intra}$ is the QP of the intra frame, $Q$ refers to the global QP value, while $Q_{offset}$ is a quantization offset ($Q_{offset} \in \mathbb{Z}$). Therefore, the solution consists in assigning a negative value to the $Q_{offset}$. The $Q_{offset}$ has been empirically fixed, all $Q_{offset}$ values in the range of $[-6,\ 0]$ have been tested and we found that the value of $-4$ is the one providing the highest coding performance. The $Q_{offset}$ applied to the CV offers an enhancement of 0.19 $dB$ in terms of BD-PSNR and $-11.7$ % in terms of BD-BR compared to the

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | 0.000 | 0.980 | | | 0.570 | -0.060 | |
| 0.370 | -0.890 | 0.000 | -1.230 | -0.700 | 0.000 | 0.001 | 0.330 |
| -0.780 | -0.400 | -0.820 | 0.490 | 1.400 | 0.730 | -0.110 | 0.360 |
| | -0.750 | 0.850 | | | 1.470 | -0.700 | |
| | 0.000 | 0.360 | | | 0.310 | 0.000 | |
| 0.470 | -0.590 | -0.650 | -0.480 | 0.330 | -0.920 | -1.010 | -0.900 |
| 1.210 | -0.580 | -0.360 | 1.041 | 0.230 | -0.310 | 1.350 | -0.670 |
| | 0.001 | 0.220 | | | 0.000 | 0.000 | |

(a)

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | 0.204 | 0.192 | | | -0.647 | -0.401 | |
| 0.391 | -0.087 | -0.254 | 0.345 | -0.154 | -0.845 | -0.477 | 0.240 |
| 0.478 | -0.044 | -0.039 | -0.132 | 0.542 | -0.781 | -0.312 | 0.310 |
| | -0.648 | 0.597 | | | 0.350 | -0.281 | |
| | -0.729 | 0.319 | | | 0.433 | -0.576 | |
| -0.371 | -0.465 | -0.439 | -0.212 | -0.701 | -1.221 | -1.020 | -0.540 |
| -0.234 | -0.303 | 0.000 | 0.556 | -0.053 | -0.936 | -0.550 | -0.109 |
| | 0.157 | 0.465 | | | -0.817 | -0.418 | |

(b)

Figure 3.7: (a) PSNR difference of the LA estimated and CNN synthesized views against the reference views of the LF images at quantization parameter (QP)=22 (negative value notice that the CNN is better then LA for this view, positive value notice the LA is better): (a) *Stairs* and (b) *University*.

H2DC-CNN coding scheme under test conditions described in Section 3.4.

### 3.3.4 Proposed Rate Distortion Optimization

As mentioned in Section 3.2, we proposed two ways to reconstruct the intermediate views ($S_I$), using LA- or CNN-based approaches. After an extensive experimentation, we found that some views are better reconstructed with LA approach rather than CNN, while for other views, the CNN approach gives better results. Figure 3.7 illustrates an example of the PSNR difference of the views linearly approximated and synthesized by the CNN block, respectivley, against the reference views. To select the right approach (LA vs. CNN), we proposed to perform a RDO for each intermediate view, thus indicating which method between LA and CNN can provide the highest RD performance, we use Algorithm 1.

**Algorithm 1:** Algorithm of the RDO between LA and CNN

```
foreach intermediate view do
    Compute Cost_RD (J) per itermediate view for both LA and CNN;
    /* Choose the best between LA and CNN for current view        */
    if Cost_RD(CNN) < Cost_RD(LA) then
        flag = 0;
    else
        flag = 1;
        X= Linear_Approximation( current_view );
        Encode X;
```

To do this, the encoder computes the Rate Distortion (RD) cost function $J$ given by Equation (5.5) for both the linearly approximated view and the one synthesized by the CNN.

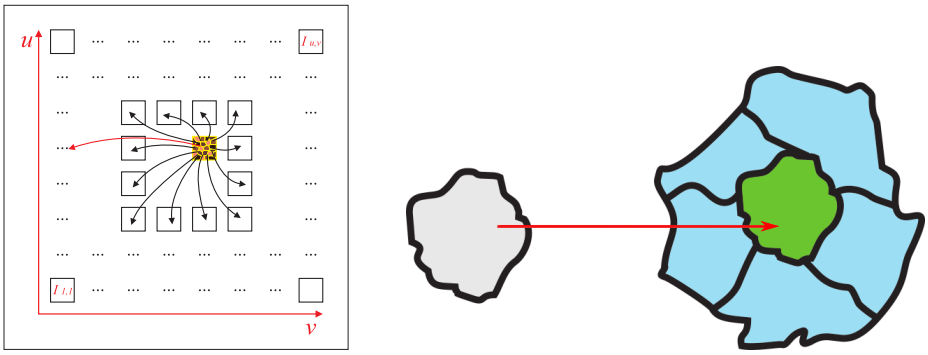
$$J = D + \lambda R \tag{3.2}$$

where $\lambda$ is the Lagrangian multiplier, $D$ is the distortion and $R$ is the rate in bits per pixel (bpp). To set the Lagrangian multiplier ($\lambda$), we empirically determine its value by testing a large set of LF images. We found that the value of 0.1 for $\lambda$ is optimal and for which the Lagrangian optimization is giving the best performance. We then select the best approach minimizing the RD cost ($J$) for each intermediate view. Therefore, an additional bit is required per intermediate view to signal which of the two methods has been selected at the encoder side ( 0: LA is selected, 1: CNN is selected ). Obviously, when LA is selected, the linear coefficients for the corresponding estimated view ($X$) need to be transmitted to the decoder, while no additional information is required for the CNN approach.

For instance, the quality improvement is noticeable for three LF images, for which the RD optimization function is giving the best RD performance illustrated in Figure 3.6. The inclusion of the RDO improves coding performance on average by 0.3 dB and $-16.1\%$ in terms of BD-PSNR and BD-BR, respectively, with respect to the H2DC-CNN coding scheme under test conditions described in Section 3.4.

### 3.3.5 Post Processing

In order to further enhance the visual quality of the reconstructed views at the decoder side, we proposed to perform a post-processing, thus offering a high visual experience. The post-processing consists in applying the Hierarchical Superpixel-to-Pixel Dense Image Matching (HSP2P) [DSS17] technique on each approximated or synthesized view ($\hat{S}_R$ and $\hat{S}_I$). The main idea is to automatically establish dense correspondences between two views in a hierarchical superpixel-to-pixel manner. Since we proposed to encode the CV at high quality, consequently, we consider it as the reference view for all the target views, as illustrated in Figure 3.8(a). Then, we partition the target views into superpixels by using SLIC method described in Section 2.2 [ASS+12], in order to find the corresponding matching superpixel for each couple of views (i.e., between the reference




(a) (b)

Figure 3.8: (a) HSP2P correlation between views, (b) Pixel in the target super-pixel $S_i^A$ (in gray) is compared with all pixels in the reference super-pixel $S_j^B$ in green and its neighbors in blue.

CV and target views). The concept of the SLIC method is illustrated in Figure 3.8(b). The feature distance is composed from the average of Lab color space and the average of Scale-Invariant Feature Transform (SIFT) feature descriptor [Low04]. To combine these two kinds of features, we define a distance function $D$ for each superpixel pair $(i, j)$ as

$$D(i,j) = \alpha_1 \| f_{lab}^i - f_{lab}^j \|_2 + \alpha_2 \| f_{sift}^i - f_{sift}^j \|_2 \tag{3.3}$$

where $(f_{lab}^i, f_{sift}^i)$ are the features of superpixel $i$ corresponding to Lab color space and SIFT feature descriptor respectively, $\| \ \|_2$ is the $L2$-norm (See Section 2.2), and ($\alpha_1$, $\alpha_2$) are two constants set to 1 and 5, respectively [DSS17].

To find the superpixel in the reference view $B$ for each superpixel in the target view $A$, a matching function M given by Equation (3.4) is computed.

$$M(i) = \arg\min_{j \in S^B} D(i,j), \quad i \in S^A \tag{3.4}$$

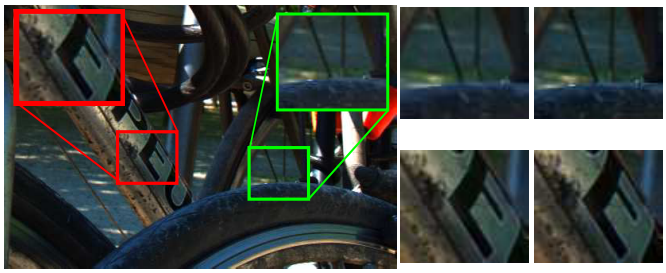
where $S^A$ and $S^B$ are the superpixel sets of the target view $A$ and the reference view $B$, respectively.

A consistency function used to calculate a coherence error for a group of matches. The domain of function is transformed into vectors TS of a superpixel $i \in S^A$ and its neighbors set N(i), which is the set of superpixels connected to superpixel i in boundaries. Given a superpixel $i$ and its corresponding matched superpixels M(i), TS(i)

$$TS(i) = c(M(i)) - c(i), \tag{3.5}$$

where c is the geometric center of a superpixel (i.e., the average coordinate of pixels in this superpixel). Then, this consistency function can be formulated as equation 3.6

$$C(i) = \frac{1}{\sum_{j \in N(i)} w_{i,j}} \sum_{j \in N(i)} w_{i,j} \| TS(i) - TS(j) \|_2 \tag{3.6}$$



(a) (b) (c)

Figure 3.9: Quality comparison for LF image *Bikes* (view (2,2)) at QP=28. a) original view, b) before post processing (PSNR = 36.22 dB, SSIM = 0.88), c) after post processing (PSNR = 36.5 dB, SSIM = 0.890).

where $w_{i,j} = exp(-\beta \| f^i_{lab} - f^j_{lab} \|_2)$ is a weighting function, which measures the similarity between two feature vectors and $\beta$ is a constant fixed as 0.02 [DSS17].

More precisely, the nearest neighbor $T(u)$ of each pixel $u$ in the target superpixel $S^A$ is searched in the reference superpixel $S^B$, based on Equation (3.7).

$$T(u) = \arg \min_{v \in cand_i} \| f^u_{lab} - f^v_{lab} \|_2, \quad \forall u \in S^A_i, \quad \forall i \in S^A \tag{3.7}$$

where $f^u_{lab}$ is the color feature using Lab color space in the updated target image and $cand_i$ represents the candidate set of superpixel $i$.

Such matching allows to improve the visual quality of the LF reconstructed image, as it refines the values of the corresponding matched pixels in the generated views. Figure 3.9 illustrates the visual quality of Bikes LF image (view (2,2)) before and after the superpixel to pixel post-processing. This post-processing enhances the coding performance by 0.17 dB and $-14.1\%$ in terms of BD-PSNR and BD-BR, respectively, compared to the H2DC-CNN coding scheme under test conditions described in Section 3.4.

It should be noted that the three previously claimed gains in each subsection are not cumulative and the overall gain of the three improvements together is given in the next section.

## 3.4 Experimental Results

### 3.4.1 Experimental Setup

**Training Phase:** For training the CNN, we run the training of DL that uses the disparity and color estimation components in two sequential CNNs. These CNNs are used to synthesize the novel views for each GOV separately with 7 layers (4 convolutions with kernel size $7 \times 7$, $5 \times 5$, $3 \times 3$, $1 \times 1$, respectively and 3 ReLUs), angular resolution $4 \times 4$ and the numerical evaluation and the final image has index (2,2). We take the 4 corner source views as input as shown in the Figure 3.11. For this training, the CNN block is trained with 100 LF images, 28 from Stanford Lytro LF dataset [RMS16],

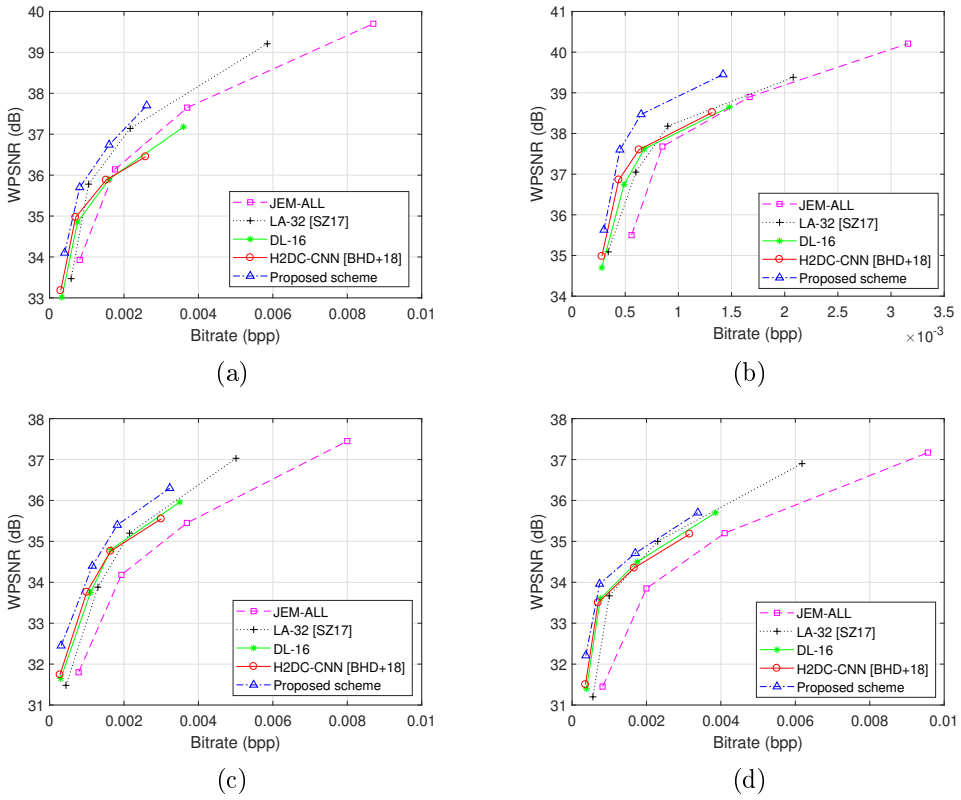



Figure 3.10: R-D curves based on wPSNR of the four considered solutions for four test LF images: (a) Building, (b) Friends1, (c) Stairs and (d) University.

and 72 from California Lytro LF dataset [KWR16], both captured by Lytro camera. We split each sub-aperture view into patches of size 60×60. This results in more than 100,000 patches which are used to train the CNN block. For more details on the training process, the reader is referred to [BHD+18].

**Testing phase:** For the testing, we select 12 LF images from two datasets of LF images captured with a Lytro Illum camera, the EPFL LF and the INRIA dataset [RE16, RSMG18], which are composed of 8×8 sub-aperture views. We use the JEM software as 2D video encoder to encode the set of 9 reference views ($S_R$) in Random Access (RA) coding configuration at 4 QP values ($QP \in \{22, 27, 32, 37\}$). We compared the proposed scheme with four state-of-the-art methods: 1) JEM-All that encodes all views with the JEM software in RA coding configuration, 2) H2DC-CNN coding scheme Section 3.2, 3) LA-32 solution [ZC17] that encodes half of the views with JEM and linearly approximates the other half, and 4) DL-16 scheme that encodes 16 views with the JEM

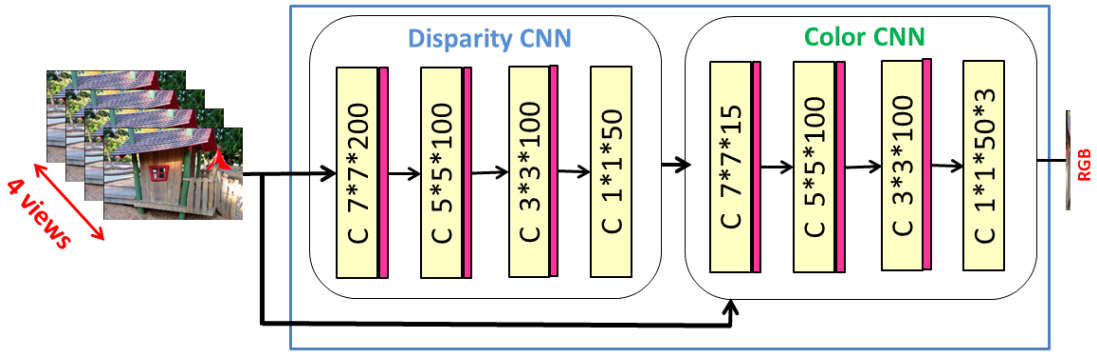



Figure 3.11: Disparity estimator neural network and color predictor neural network consists of four convolutional layers with decreasing kernel sizes.

and synthesizes the rest of views at the decoder.

### 3.4.2 Results

### 3.4.3 Objective Evaluation

R-D curves based on WPSNR for four LF images are provided in Figure 3.10. We can notice that the proposed scheme provides for both images the highest PSNR performance at all considered bitrates. The previous conclusion is confirmed by Table 3.1, providing the Bjøntegaard results [hB08] of the four considered solutions, compared to the anchor solution JEM-All for the whole set of LF images. Our proposed method achieved an average BD-BR gain of −50.34% and BD-PSNR of 1.393 dB compared to the JEM-All solution. We can also notice that the proposed solution achieves a gain for all considered LF images including Bee2 for which the BD-BR ranges from loss of

Table 3.1: BD-BR and BD-PSNR gains calculated against anchor JEM-All for 12 LF images. 1) Bikes 2) Friends1 3) Friends4 4) Rolex 5) RustyFence 6) Stairs 7) University 8) FountainVincent2 9) YanKriosStanding 10) Bee2 11) Building 12) Cactus.

| | LA-32 [ZC17] | | DL-16 | | H2DC-CNN [BHD+18] | | Proposed scheme | |
|---|---|---|---|---|---|---|---|---|
| Im. | BD-BR | BD-PSNR | BD-BR | BD-PSNR | BD-BR | BD-PSNR | BD-BR | BD-PSNR |
| *1)* | -29.17% | 0.85 | -36.36% | 0.93 | -37.39% | 0.73 | **-46.15%** | **1.19** |
| *2)* | -26.31% | 0.57 | -27.58% | 0.50 | -36.61% | 0.76 | **-51.84%** | **1.51** |
| *3)* | -28.65% | 0.66 | -24.73% | 0.42 | -34.10% | 0.66 | **-57.83%** | **1.84** |
| *4)* | -20.77% | **0.45** | -11.46% | -0.10 | -11.20% | -0.29 | **-30.29%** | 0.12 |
| *5)* | -24.71% | 0.70 | -31.15% | 0.61 | -37.39% | 0.66 | **-48.39%** | **1.22** |
| *6)* | -29.50% | 0.83 | -42.28% | 1.12 | -44.68% | 1.19 | **-52.42%** | **1.64** |
| *7)* | -30.57% | 0.81 | -45.92% | 1.13 | -46.88% | 1.10 | **-54.73%** | **1.50** |
| *8)* | -28.07% | 0.66 | -37.65% | 0.73 | -37.83% | 0.52 | **-46.81%** | **0.95** |
| *9)* | -24.26% | 0.78 | -57.32% | 1.94 | -62.25% | 2.21 | **-69.63%** | **2.87** |
| *10)* | -2.81% | 0.04 | -1.14% | -0.27 | 7.76% | -0.54 | **-36.90%** | **0.30** |
| *11)* | -22.89% | 0.53 | -9.41% | 0.04 | -19.80% | 0.20 | **-38.29%** | **0.83** |
| *12)* | -15.94% | 0.46 | -54.07% | 1.55 | -59.96% | 1.83 | **-70.77%** | **2.70** |
| Av. | -23.63% | 0.61 | -31.58% | 0.71 | -35.02% | 0.75 | **-50.34%** | **1.393** |

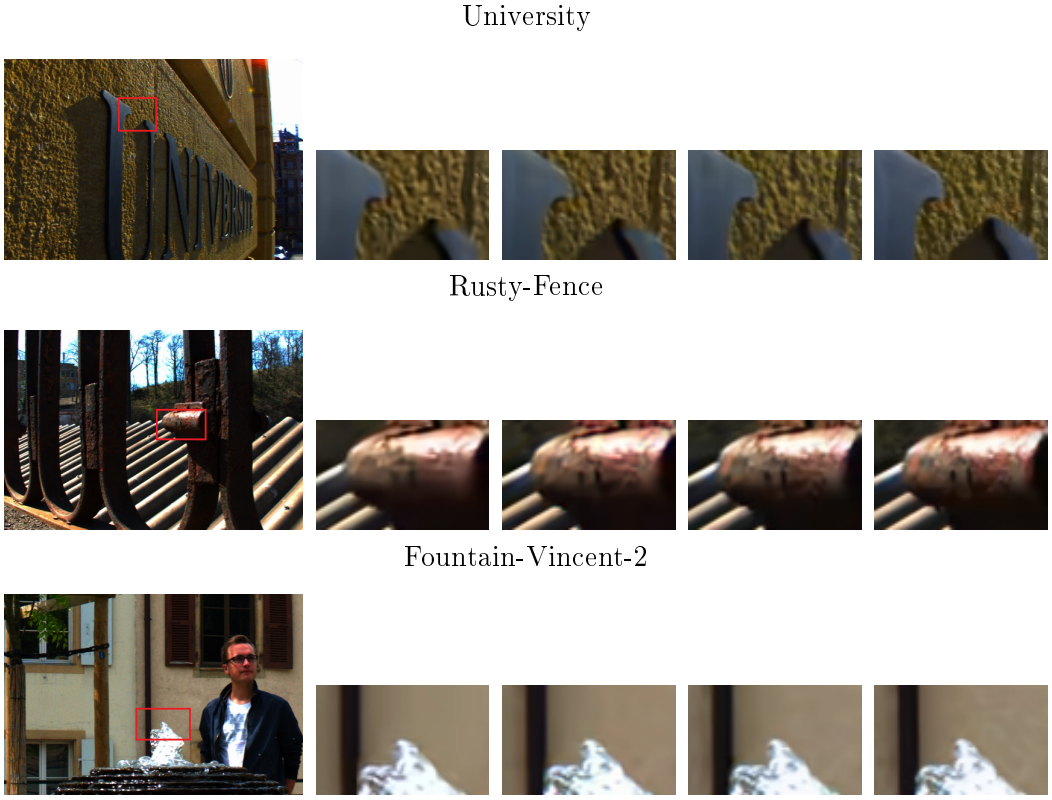



Figure 3.12: Overall visual comparisons for showing the visual quality of view at position $(3, 2)$ of the 4 methods: a) original views, cropped decoded views by b) JEM-All, c) LA-32, d) DL16 and e) our proposed method.

7.76 % for H2DC-CNN coding scheme to a gain of $-36.90\%$ for the proposed solution.

Figure 3.12 shown the visual quality of the two LF images with the four considered solutions. *University*, b) bitrate= 0.01790 bpp, WPSNR= 32.10 dB, SSIM= 0.788, c) bitrate= 0.01664 bpp, WPSNR= 33.38 dB, SSIM= 0.829, d) bitrate= 0.01610 bpp, WPSNR= 33.42 dB, SSIM= 0.837, e) bitrate= 0.01619 bpp, WPSNR= 33.81 dB, SSIM= 0.85. $Rusty - Fence$, b) bitrate= 0.01730 bpp, WPSNR= 31.91 dB, SSIM= 0.866, c) bitrate= 0.01649 bpp, WPSNR= 33.465 dB, SSIM= 0.901, d) bitrate= 0.01577 bpp, WPSNR= 33.41 dB, SSIM= 0.905, e) bitrate= 0.01710 bpp, WPSNR= 34.29 dB, SSIM= 0.935. $Fountain_Vincent_2$, b) bitrate= 0.01670 bpp, WPSNR= 34.97 dB, SSIM= 0.877, c) bitrate=0.01654 bpp, WPSNR= 36.43 dB, SSIM= 0.901, d)bitrate= 0.01601 bpp, WPSNR= 36.447 dB, SSIM= 0.907, e) bitrate= 0.01597 bpp, WPSNR= 36.68 dB, SSIM= 0.913.

Table 3.2 reports the performance in terms of BD-BR based on SSIM [ZBSS04], comparing the proposed scheme with the anchors JEM-All and H2DC-CNN. It is clear that the proposed scheme provides better results than both H2DC-CNN and JEM-ALL,

Table 3.2: Coding gains of the proposed solution in BD-BR based on SSIM and PSNR.

| | PSNR-based | | SSIM-based | |
|---|---|---|---|---|
| | BD-BR | BD-PSNR | BD-BR | BD-SSIM |
| vs JEM-All | -50.34% | 1.393 | -63.71% | 0.031 |
| vs H2DC-CNN | -30.06% | 0.619 | -21.14% | 0.008 |

where the gain in terms of BD-BR is about $-63.71\%$ and $-21.14\%$ compared to JEM-All and H2DC-CNN, respectively.

### 3.4.4 Time Complexity

Recently, the number of large-scale applications of accounts is constantly increasing, especially with deep machine learning. The number of cores in the Central Processing Unit (CPU) is much lower than the Graphics Processing Unit (GPU). The GPU consists of hundreds of small cores capable of simple calculations. The degree of parallelism and speed of execution are problems of low data volume on the CPU. Then, GPU is fit for training the deep learning systems in a long run for very large datasets.

The complexity of the proposed scheme is also evaluated and compared to the other methods on both CPU and GPU platforms. The performance has been carried-out on an Intel core i9-7900X CPU running at 3.3GHz PC with 64 GB memory and a TITAN Xp NVDIA GPU. It is important to note that the GPU is only used when the CNN block is involved in the coding scheme. Table 3.3 gives the encoding and decoding run times in seconds. We can notice that the proposed solution achieves the fastest encoding at all QP, and the GPU enables to speedup the encoding part related to the CNN block. However, the decoder of the proposed solution is complex mainly due to the post-processing stage that significantly increases the decoding complexity.

Table 3.3: Running time in seconds of the encoder and decoder for *Bikes* image.

| | Encoder side | | | | Decoder side | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | JEM-All | LA-32 [ZC17] | Our | | JEM-All | LA-32 | DL-16 | | Our | |
| QP | CPU | CPU | CPU | GPU | CPU | CPU | CPU | GPU | CPU | GPU |
| 22 | 1369 | 1039 | 727 | **650** | **4** | 11 | 114 | 69 | 325 | 279 |
| 26 | 1086 | 850 | 669 | **600** | **3** | 10 | 113 | 67 | 325 | 277 |
| 32 | 778 | 675 | 592 | **521** | **3** | 10 | 113 | 67 | 323 | 276 |
| 37 | 599 | 572 | 535 | **462** | **3** | 9 | 113 | 65 | 322 | 274 |

## 3.5 Conclusion

In this chapter, we have presented our preposed LF coding scheme, where a set of views are taken as reference, while the other set of views is estimated. In particular, our coding scheme performs fine-tuning of the CV quality, which is used as a reference by the rest of the LF views. We uses local RDO functionality, that allows to choose the best coding way between LA and CNN for each intermediate view. Finally, with the aim to

further enhance the quality of the reconstructed LF views, a super-pixel to pixel dense correspondence is carried out as a post-processing. The enhanced proposed scheme increases the coding efficiency by 30.06% compared to the state-of-the art solutions, while providing LF images with high visual quality.

.

# Chapter 4

# Subjective Evaluation of Light Field Image Compression Methods Based on View Synthesis

## 4.1 Introduction

Subjective quality evaluation of images and videos is a very active field of research. In this chapter, we propose to conduct subjective experiments of LF compression methods based on view synthesis technique. Specifically, four compression approaches have been considered in this study, two methods are view synthesis basis, while the remaining are naive LF coding methods. All these methods have been subjectively and objectively evaluated. The dataset, including non-compressed and compressed LF images, along with subjective scores are provided publicly to facilitate future research works, such as developing new reliable objective quality metrics for LF images based view synthesis methods.

This chapter is organized as follows. Section 4.2 describes the performed subjective experiment, including the preparation of the test material, environmental setup and the test methodology. Section 4.3 presents the LF coding methods considered in this study. Section 4.4 described the subjective evaluation. Next, the results and analysis of subjective evaluation are provided in Section 4.5. Finally, Section 4.6 concludes the chapter.

## 4.2 Environment Setup and Test Methodology

A total of 18 naive subjects (10 females and 8 males) took part in the subjective experiments. The age of subjects was ranging from 20 to 58, with an average of 29.4. All subjects were screened for color blindness and visual acuity using Ishihara and Snellen charts, respectively.

The subjective evaluations were conducted in a laboratory psychovisual test room, calibrated according to ITU-R BT.500-13 Recommendations [BT.12b], equipped with

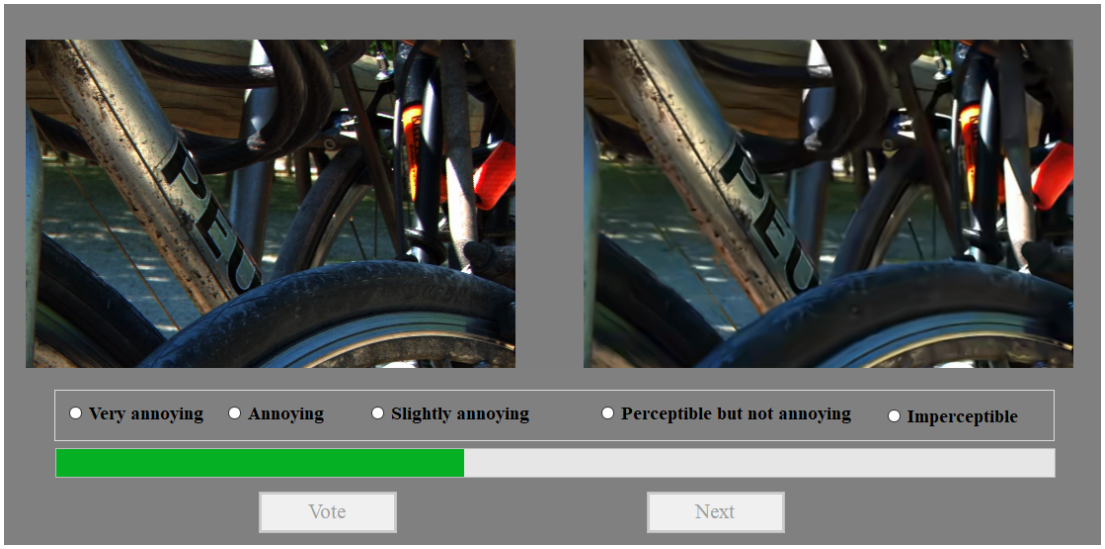



Figure 4.1: Screenshot from the subjective study interface displaying the video to the subjects.

a controlled lighting system and the color of all background walls and curtains is mid-gray. A full HD 27-inch Dell UltraSharp U2717D was used to display the test stimuli. The distance of the subjects from the monitor was approximately equal to 7 times the image height, as recommended in [BT.12a].

The subjective experiments have been performed using the recently introduced methodology, named passive test methodology [VrE17], without refocusing effect. The methodology is based on DSIS [BT.12b], where both the non-compressed reference and stimulus were displayed in a side-by-side arrangement on the same monitor (described in Section 2.7.2). The non-compressed reference and stimulus were always displayed on the left and right side, respectively, and the subjects were aware of these positions, as shown in Figure 4.1. In addition, the LF contents were presented as a video sequence navigating between the viewpoints. The pseudo-video was created using horizontal scan, starting from the view in the left upper corner down, and proceeding from left to right and right to left in alternate order, which mimics the parallax effect. In [BCC18], it has been noticed that this visualization technique is preferred among six possible different visualization strategies, because it reduces the shift among consecutive frames. Moreover, the created videos were displayed with a frame rate of 9 frames per second offering a smooth switching between views.

At the end of the presentation of each pair of videos, a dedicated user interface was displayed on the screen for about five seconds during which the subject gives its judgment. The participants were asked to rate the level of impairment of the stimulus with respect to the non-compressed reference, using a five-grade discrete impairment scale (1: very annoying, 2: annoying, 3: slightly annoying, 4: perceptible, but not annoying, 5: imperceptible).

Given the large number of stimuli, a session would exceed 30 minutes, making it hard to show all of them in a single session. Consequently, in order to avoid visual fatigue effects, the subjective experiment was divided into two sessions whose duration does not exceed 20 minutes each. Subjects took a break between each two sessions. Moreover, each test session involved only one subject assessing the stimuli. In order to avoid possible contextual and memory effects, the display order of these stimuli was randomized in a way that the same content was never shown consecutively.

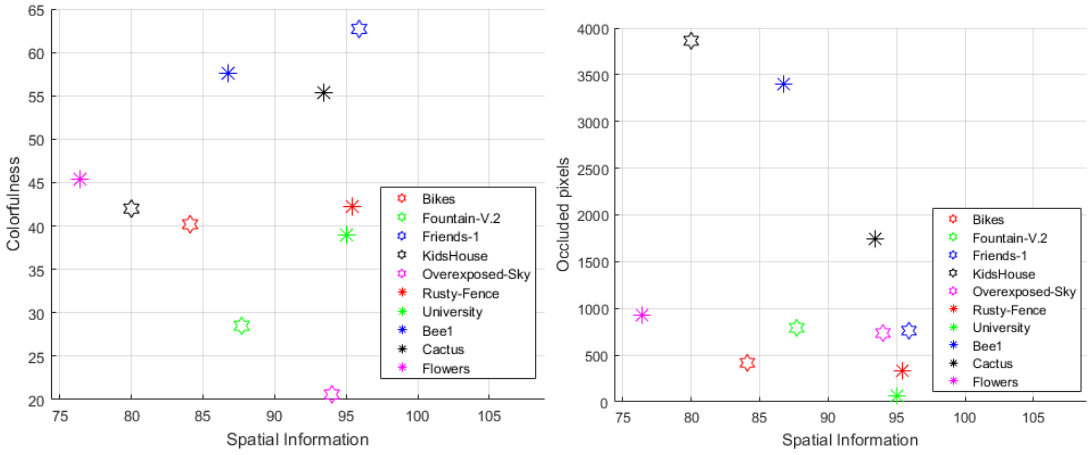
Before the experiment starts, instructions explaining the task were provided to subjects. In addition, training session was held with additional LF contents, allowing the subjects to practice and become familiarized with the test procedure. The quality of these training samples was chosen so that it covers the full rating scale.

## 4.3 Evaluated Light Field Coding Strategies

The LF contents evaluated in the subjective experiments were compressed using four coding strategies. Given that the widely explored coding approach for LF contents is the pseudo-video sequence coding method, we have therefore considered two methods from this category. For both coding methods, all the sub-aperture images are rearranged into a pseudo-sequence using spiral order scan starting from the center view, which is then encoded with a classical video encoder. Two video encoders have been selected for this purpose, the HEVC standard and the JEM that led to the starting point of future video coding standard named VVC. For HEVC, the HM reference software (version 16.9) was used, while for the second method the JEM software (version 7.0) was exploited, both in random access coding configuration. For both methods, all views are encoded and we refer to them as HM-All and JEM-ALL for the rest of this chapter. In addition, in order to avoid the darkness and distorted remote views, only the middle $8 \times 8$ views were encoded.

Furthermore, two Light Field compression methods based on view synthesis have been included in this study. Instead of coding all views, in these approaches, only sparse samples of LF views are encoded and transmitted, while the other views are synthesized at the decoder side. One of the selected methods is described in [ZC17], where at the encoder side the views are equally divided into two sets, the selected reference views set and the dropped views set, that is 32 views each. The selected reference views are then rearranged into a pseudo-sequence using horizontal zigzag scan order and compressed with a 2D video encoder standard (JEM in our implementation). The decoded versions of theses latter views are used to linearly approximate the dropped views and only the approximation coefficients are transmitted to the decoder. At the decoder side, the selected reference views are decoded and the dropped views are approximated by the weighted sum of the decoded selected views. For the rest of this chapter, we refer to this method as LA-32.

Finally, the fourth and last method that we included is the CNN-based view synthesis approach proposed in [HASM18]. In this method, the authors proposed a learning-based approach to synthesize new views from a sparse set of input views. The proposed




(a) (b)

Figure 4.2: Distributions of the three properties of the selected LF contents.

architecture includes two phases: a disparity estimator and color predictor, which are performed by two sequential CNNs. Based on the features extracted from the sparse input views (four views at the corners), four layers CNN firstly estimates the disparity of the dropped views. The second CNN uses all the warped disparity views, derived from the first CNN, along with few other features to predict the color and synthesize the dropped views. For training the CNN, we used 100 LF images, 28 from Stanford Lytro LF dataset [RMS16] and 72 from California Lytro LF dataset [KWR16]. We split each sub-aperture view into patches of size $60 \times 60$, which results in more than 100,000 patches exploited for training. For this method, which will be referred to as DL-16, 16 sparse views are encoded with the JEM, while the remaining dropped views are synthesized by the trained CNN block at the decoder side.

## 4.4 Subjective Evaluation

### 4.4.1 Dataset Preparation

**Scene characteristics:** The selection of test scenes is an important issue. In particular, the spatial and color features and the amount of occluded pixels of the scenes are critical parameters. The dataset must contain images with various charactersitics as possible.

**Spatial perceptual information measurement:** The spatial perceptual information (SI) is based on the Sobel filter. Each video frame (luminance component) at time n ($F_n$) is first filtered with the Sobel filter [Sobel($F_n$)]. The standard deviation over the pixels ($std_{space}$) in each Sobel-filtered frame is then computed. This operation is repeated for each frame in the video sequence and results in a time series of spatial information of the scene [P.908]. The maximum value in the time series ($max_{time}$ ) is

chosen to represent the spatial information content of the scene. This process can be represented in equation form as:

$$SI = max_{time}\{std_{space}[Sobel(F_n)]\} \tag{4.1}$$

**Colorfulness measurement:** Colorfulness, referred also as chromaticness, is the attribute of a visual sensation according to which the perceived color of an area appears to be more or less chromatic.

The definition of colorfulness is very similar to chroma, but chroma is relative perception. Colorfulness usually increases as the luminance is increased, except when the brightness is very high (very colorful outdoor images). Colorfulness of the stimulus is its measure of the intensity of the hue. For measure of colorfulness we should examine the presence of high-saturation colors along various hues [DH03].

To calculate the colorfulness M, we use the following formula:

$$M = \sqrt{{\sigma_{rg}}^2 + {\sigma_{yb}}^2} + 0.3 \times \sqrt{{\mu_{rg}}^2 + {\mu_{yb}}^2} \tag{4.2}$$

where: $\sigma$ and $\mu$ are the standard deviation and the mean value of the pixel cloud along direction described by subscripts and:

$$rg = R - G \tag{4.3}$$

$$yb = \frac{1}{2}(R + G) - B \tag{4.4}$$

**Amount of occluded pixels:** One of the most important characteristics of Light Field image is the occlusion exactly with sub-aperture array representation.

**Selected dataset:** In order to cover a wide range of features, the spatial complexity, color features and the amount of occluded pixels of each LF image have been analyzed using Spatial Information (SI) [P.908], ColorFulness (CF) [DH03] and occlusion model proposed in [WER16], respectively.

Based on these features, a total of ten LF images have been carefully selected for subjective experiments, six from EPFL Light-Field Image Dataset (Bikes, Fountain_&_Vincent_2, Friends-1, Overexposed-Sky, Rusty-Fence and University) [RE16], two from INRIA Light-Field Image Dataset (Bee1 and Cactus) [JPF+17] and two that we acquired by a Lytro Illum camera, namely Flowers and KidsHouse. These LF images represent different content, including indoor and outdoor scenes and a wide range of colors, textures and depth properties [PGL+17]. Figure 4.2 shows the values of SI, CF and occlusions for all the selected images.

Each image was extracted from LF raw file format using Light Field Matlab Toolbox v0.4 [DPW13], thus providing a 4D LF of dimensions $15 \times 15 \times 434 \times 625 \times 4$, where $434 \times 625$ represents the resolution of each view, 4 corresponds to the RGB channels including additional weighting image component, while $15 \times 15$ represents the number of

(a) Bikes

(b) FountainVincent2

(c) Friends1

(d) OverexposedSky

(e) RustyFence

(f) University

(g) Bee1

(h) Cactus

(i) Flower

(j) KidsHouse

Figure 4.3: The thumbnails of every LF images used for the subjective test.

views [RE16]. As mentioned previously, we only encoded the central $8 \times 8$ sub-aperture views after being converted to YUV format and downsampled to 4:2:0 with 10-bit depth. The ten LF images have been encoded using the previously described four compression methods at four compression bitrates, namely R1 = 0.0074 bpp, R2 = 0.0171 bpp, R3 = 0.0384 bpp and R4 = 0.1112 bpp.

### 4.4.2 Data Processing

First, the subjective scores were screened to detect and exclude possible outliers and we verified the distribution of individual participant scores. Indeed, some data may interfere with the results. Outliers detection was performed as specified in [BT.12b], and no outlier subjects were found in this study.

Second, the Mean Opinion Score (MOS) was computed as the mean across scores provided by different subjects as follows:

$$MOS_j = \frac{1}{N} \sum_{i=1}^{N} s_{ij} \tag{4.5}$$

where $N$ is the number of subjects and $s_{ij}$ is the score given by subject $i$ for the stimulus $j$.

In order to evaluate the reliability of the obtained results from statistical point of view, 95% confidence intervals (CI), assuming a Student t-distribution of the scores, were computed together with MOS values.

### 4.4.3 Statistical Analysis

In order to test the existing one for an influence on the judgment of participants in the quality assessment process, and to verify if this influence is statistically significant, we performed an ANOVA analysis of variance. Three parameters were introduced in this experiment: Content, video content (Content of pseudo video), QP and compression method. The degree of influence of the parameter is based on the value of $p$. The $p$-value is the level of marginal significance within a statistical hypothesis test representing the probability of the occurrence of a given event.

Table 4.1: Analysis of variance, one factor for all configurations.

| Factor | p-Value | Impact |
|---|---|---|
| quantization parameter | <0.0001 | *** |
| Compression method | 0.002 | ** |
| Content of pseudo video | 0.997084 | |

The results in the Table 4.1 showed that the quality level, expressed by this quantification parameter, has a major influence on the scores obtained ($p < 0.0001$) while the content has no influence in this case (with a $p = 0.997$)

## 4.5 Results and Discussion

R-D curves based on weighted PSNR (wPSNR) of the four evaluated methods are provided in Figure 4.4. In these plots, the horizontal axis reports the bitrate required to encode the LF image and the vertical axis represents the average wPSNR across all sub-aperture images calculated for YUV channels, where the factor 6 is assigned to the luminance channel and the factor 1 for each chrominance channel [OSS+12].

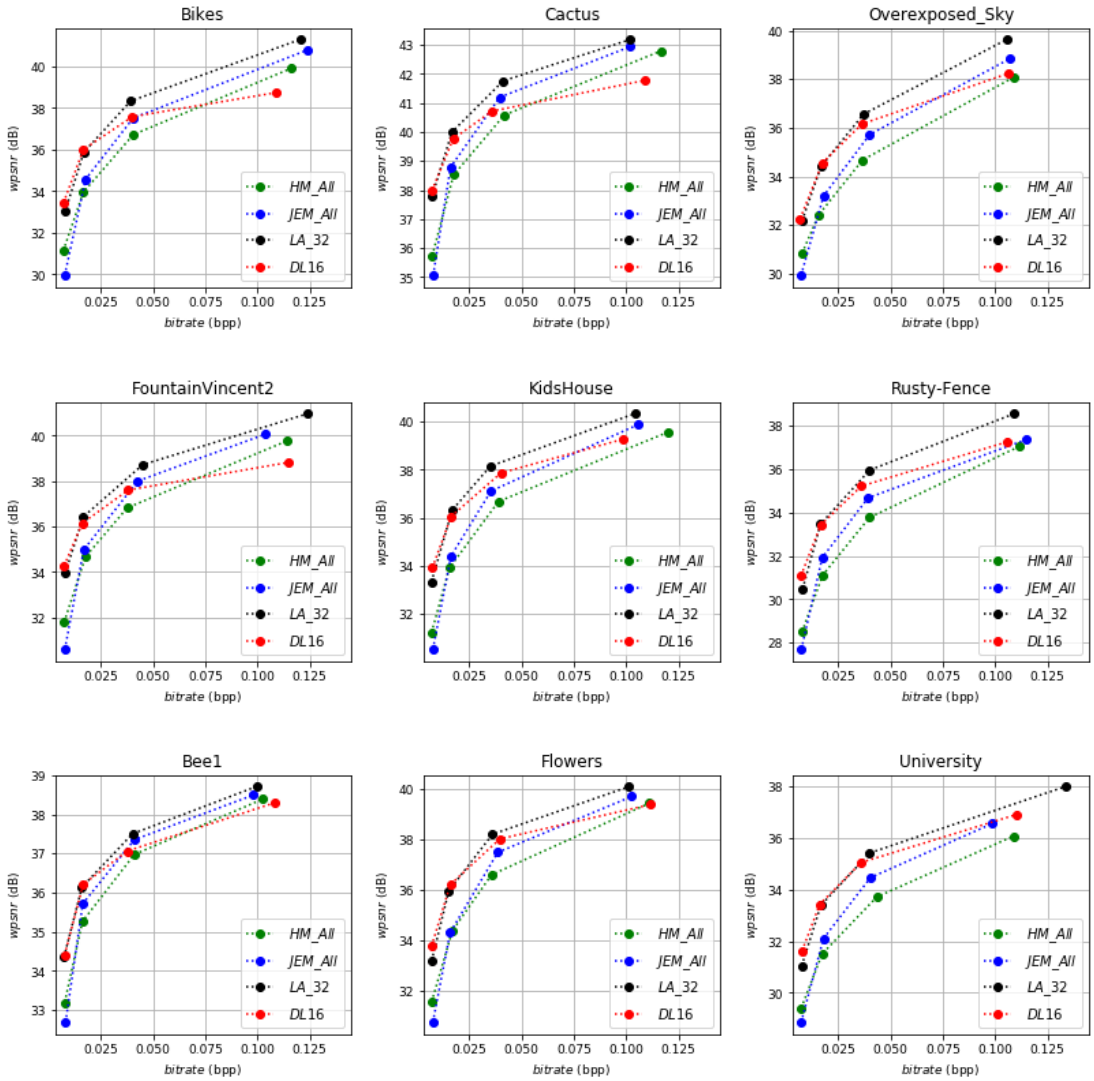



Figure 4.4: R-D curves based on wPSNR of the four considered solutions for six different LF images.

One can observe that for all LF images and for all bitrates the LA-32 method provides the best result and outperforms the other compression solutions. The CNN-based

view synthesis approach (DL-16) performs well at low and medium bitrates compared to HM-ALL and JEM-ALL methods, whereas it provides low performance for the high bitrates. As expected, JEM-ALL outperforms HM-ALL for all tested LF images and for all bitrates, because it includes different improvements compared to HM, thus leading to an improvement of R-D performance. However, these results are reported according to wPSNR objective metric, which is not the best way for assessing the visual quality of LF images.

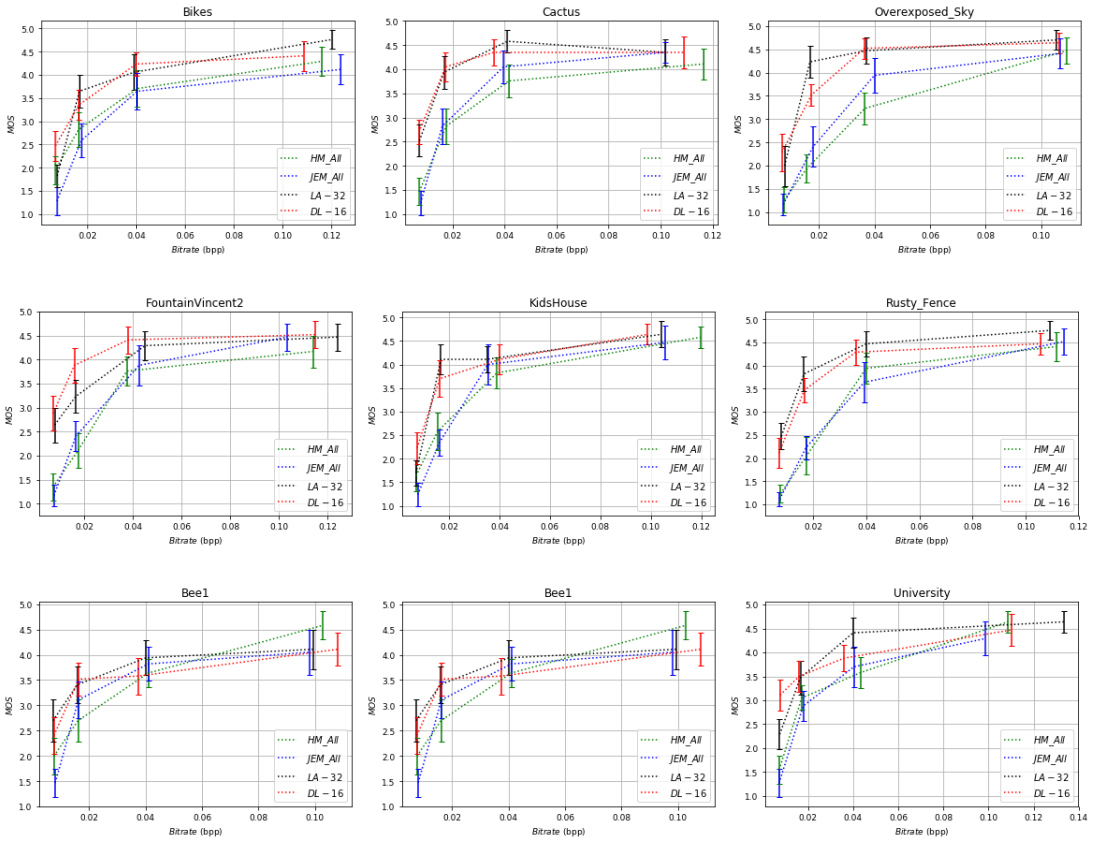
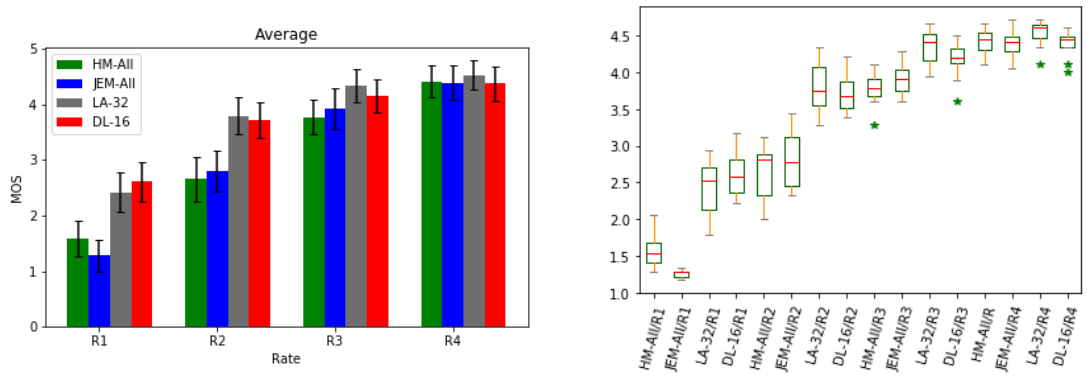



Figure 4.5: MOS vs bitrate with associated confidence intervals for six different LF images.

Thus, in Figure 4.5, the fitted R-D curves based on the MOS are illustrated. The same conclusion may be drawn from this Figure regarding the LA-32 method. However, for DL-16 method, the results are quite different from objective evaluation, since this method achieves clearly better visual quality than HM-ALL and JEM-ALL methods, especially at low and medium bitrates. Globally, the LF coding methods based on view synthesis (LA-32 and DL-16) provide the highest visual quality at all bitrates. For instance, for most LF images their visual quality provided at medium bitrate is roughly the same as the one achieved by the naive coding approaches (HM-ALL and JEM-ALL)

at high bitrate. Thus, the coding methods based on view synthesis can achieve high coding performance and demonstrate their effectiveness by providing the best visual quality compared to the two other methods.




(a) (b)

Figure 4.6: Comparison of average MOS scores for each of the methods (HM-All, JEM-All, LA-32, DL-16,) at 4 different flow rates (R1, R2, R3, R4): (a) represent the average MOS over the 10 images (Testing Set). (b) box plot showing the distribution of the MOS scores. Median in the box represented by the red line. Whiskers denote most extreme points, not considering outliers.

It is easy to see in Figure 4.6 that both methods of view synthesis-based coding (DL-16, LA-32) have a visibly high MOS average compared to JEM-ALL and HM-ALL. In addition, it can also be noted that the behaviour of view synthesis based coding methods is similar. also for the other two methods.

## 4.6 Conclusion

In this chapter, two recent LF compression methods based on view synthesis have been compared subjectively and objectively to two pseudo-video sequence based coding approaches. Experimental results show that the methods based on view synthesis achieve significant better coding performance without affecting the visual quality. Specifically, the subjective quality assessment showed that the view synthesis based methods provide substantial superior visual quality, especially at low and medium bitrates. Finally, subjective evaluation helped us to know that some coding methods visually outperforms other methods, which was not remarkable during the objective evaluation.

# Chapter 5

# Light Field Image Coding Using Dual Discriminator Generative Adversarial Network and VVC Temporal Scalability

## 5.1 Introduction

A possible extension of our proposed work in chapter 3, is to use a more advanced neural network for a better missing views synthesis. Studies proved that GAN show a great performance in this field [JZW+18]. In particular version of the GAN is the D2GAN where a double networks are used to enhance the quality of the synthesized images [NLVP17].

In this chapter, we propose an efficient approach to encode the LF images, which consists in encoding a sparse set of views, and estimate the rest of views at the decoder side. In particular, the first set of selected reference views are coded with the next generation video coding standard called VVC. While the second set of views are either synthesized from the first decoded set of views using a D2GAN or decoded by a VVC decoder. The D2GAN have been trained with a large set of LF images coded at different distortions. The architecture offered by the D2GAN, composed by a generator and two discriminators, enables better training and thus synthesizes views with high visual quality. In addition, to increase the coding efficiency, a RDO is adopted to select which views should be encoded and transmitted and which ones should be dropped and synthesized at the decoder side.

The remainder of this chapter is organized as follows. Section 5.2 describes the concepts of D2GAN and VVC. Then, in Section 5.3, we describe the proposed LF image compression solution. Section 5.4 presents and discusses the experimental results. Finally, Section 5.5 concludes this chapter.

## 5.2 Background

As mentioned in Section 5.1, the proposed coding approach is based on D2GAN and VVC standard. In this section, we briefly introduce these two concepts.

### 5.2.1 Dual Discriminator Generative Adversarial Nets

GANs are deep neural net architectures composed of two consecutive neural network models, namely generator $G$ and discriminator $D$. GAN enables to simultaneously train the two models: the generative model $G$ that captures the data distribution, and the discriminative model $D$ that estimates the probability that a sample came from the training data rather than from $G$ [Ga14]. GAN has recently achieved great successes in various fields, especially in fake video generation, super-resolution and objects detection [LTH+17, BZDG18].

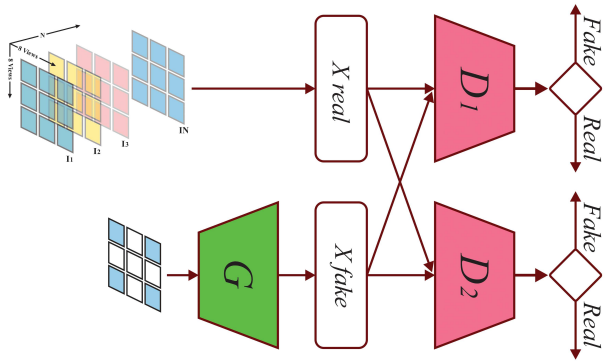



Figure 5.1: Dual discriminator generative adversarial networks architecture.

D2GAN, is a novel framework based on GAN, which uses two discriminators $D_1$ and $D_2$, where $D_1$ tries to assign high scores for real data, and $D_2$ tries to assign high scores for the fake data, as shown in Figure 5.2. This technique uses the two discriminators to minimize the Kullback-Leibler (KL) divergence and reverse KL between the generated image and the target image [NLVP17]. Formally, $D_1$, $D_2$ and $G$ now play the following three player minimax optimization game:

$$\begin{aligned}\min_{G}\max_{D1,D2} \jmath\,(G, D1, D2) &= \alpha\,\mathbb{E}_{x\backsim P_{data}}[\log D_1(x)] \\ &+ \mathbb{E}_{z\backsim P_z}[-D_1(G(z))] + \mathbb{E}_{x\backsim P_{data}}[-D_2(x)] \\ &+ \beta\,\mathbb{E}_{z\backsim P_z}[\log D_2(G(z))],\end{aligned} \tag{5.1}$$

where $z$ is a noise vector, $\mathbb{E}$ represents expected value, $x$ is the real data, $P$ represents the probability distribution, $\alpha$ and $\beta$ are two hyper-parameters ($0 < \alpha$, $\beta \leq 1$) to stabilize the learning of the model and control the effect of KL and reverse KL divergences on the optimization problem [NLVP17].

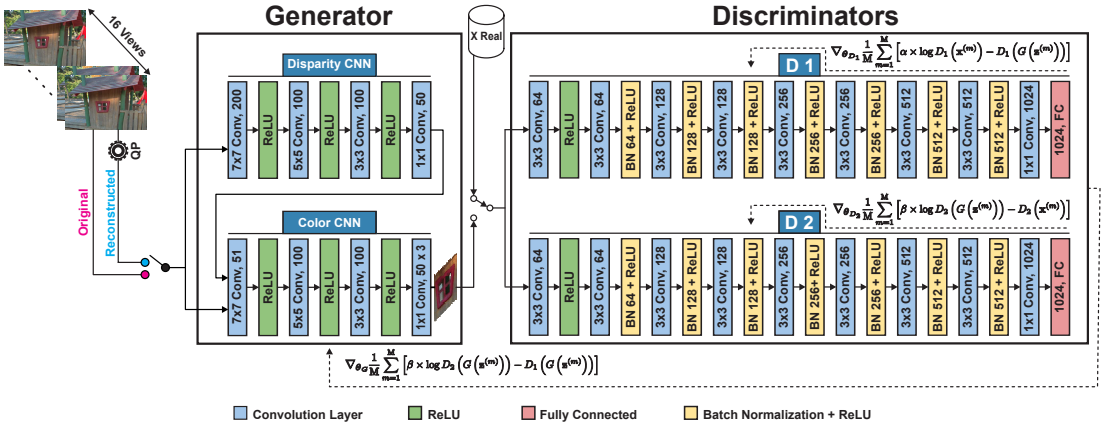



Figure 5.2: Detailed D2GAN architecture.

More specifically, with a batch of $M$ noise samples $z^{(1)}, z^{(2)}, ..., z^{(M)}$ given as inputs, the generator generates $M$ artificial samples, and this process is defined as $G(z^{(i)})$. While, $x^{(1)}, x^{(2)}, ..., x^{(M)}$ represents a batch of $M$ real data samples.

Three cost functions defined in (5.2), (5.3) and (5.4) are computed to obtain the error that should be transmitted respectively to $D_1$, $D_2$ and $G$ for their backward weights updating, as shown in Figure 5.2 (dash lines).

$$\nabla_{\theta_{D1}} \frac{1}{M} \sum_{m=1}^{M} [\alpha \log D_1(x^{(m)}) - D_1(G(z^{(m)}))], \tag{5.2}$$

$$\nabla_{\theta_{D2}} \frac{1}{M} \sum_{m=1}^{M} [\beta \log D_2(G(z^{(m)})) - D_2(x^{(m)})], \tag{5.3}$$

$$\nabla_{\theta_G} \frac{1}{M} \sum_{m=1}^{M} [\beta \log D_2(G(z^{(m)})) - D_1(G(z^{(m)}))]. \tag{5.4}$$

In this work, we use D2GAN to synthesize the dropped LF views, where the generator consists of two CNN [KWR16], the first CNN estimates the disparity and the second one generates the color image.

### 5.2.2 Versatile Video Coding

Based on HEVC, Joint Video Exploration Team (JVET) is developing a new video coding standard called VVC [MWS17]. VVC already enables a bitrate saving of 35% to 40% with respect to HEVC for the same visual quality [SHD+19]. VVC introduces several new coding tools at different levels of the coding chain including frame partitionning, intra/inter predictions, transform, quantization and entropy coding. For more details about the VVC coding tools the reader can refer to [RHPD19]. VVC supports by design the temporal scalability through the RA coding configuration. This latter,

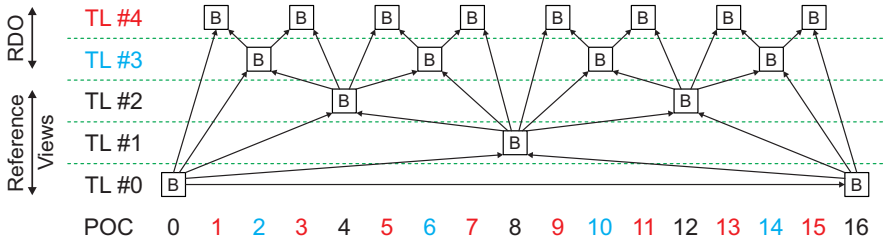



Figure 5.3: Hierarchical prediction structure in VVC. One GOP is shown.

illustrated in Figure 5.3, enables different temporal layers and each temporal layer uses as reference only frames from lower temporal resolution, i.e., lower layer. Therefore, frames of each temporal layer $t_i$ can be removed without impacting the decoding of frames of lower temporal resolution $t_j$ with $t_i > t_j$.

In the proposed coding approach, we exploit the concept of temporal resolution to drop views at the encoder without impacting the decoding process and thus performing the best rate distortion performance.

## 5.3 Proposed LF Image Compression Method

The idea behind the proposed coding method is, instead of transmitting all the LF views, to drop a sub-set of views at the encoder side and synthesize them at decoder side, thus considerably reducing the required bitrate for LF images. To efficiently achieve that, we exploit the temporal scalability of VVC and use the D2GAN model, all in a RDO process.

At the encoder side, first, LF sub-aperture views are organized into groups of 16 views that form GOPs, as illustrated in Figure 5.3. Next, in each GOP, the images of temporal levels 0, 1 and 2 are encoded using the VVC codec, which constitute the reference views used later in the synthesis process at the decoder side. Then, the images at the remaining levels 3 and 4 are either coded using the VVC codec or dropped. In contrast to fix the number of dropped views, in our approach this is done adaptively on the basis of the proposed RDO process described in the Algorithm 2 and explaining in the following.

As illustrated in Figure 5.3, we apply RDO process on the 3 consecutive frames, i.e., frame $i$ at level 4, frame $i+1$ at level 3 and frame $i+2$ at level 4. It should be noted that if one of the views at temporal level 4 (frame $i$ or $i+2$) must be encoded using VVC, then the frame $i+1$ at level 3 is also encoded using VVC, because it will be used as a reference for the frames at temporal level 4.

Main reasons behind only considering the 2 upper levels exclusively to the RDO block are, firstly, after an extensive study, we found that these levels occupy together around 28% of the total bitrate. Second, the views at the upper levels are not used as references in the VVC coding scheme.

Thus, we proposed a RDO block deciding which views from the upper level can be encoded using VVC or dropped and synthesized using D2GAN. To reach this goal, the

**Algorithm 2:** Algorithm of the RDO between VVC and D2GAN

**Require:** $\mathcal{J} \leftarrow \{ \forall\, m,\ \forall\, v \in TL\#[3\ or\ 4],\ \mathcal{J} = D + \lambda\, R\}$
m: metod {VVC, D2GAN}
**for all** $v \in TL\#4$ **do**
  **if** $\mathcal{J}(VVC) < \mathcal{J}(D2GAN)$ **then**
    Encode $v$ by VVC
    flag($v$) $\leftarrow$ false
  **else**
    generate $v$ by D2GAN
    flag($v$) $\leftarrow$ true
  **end if**
**end for**
**for all** $v \in TL\#3$ **do**
  **if** $\mathcal{J}(VVC) < \mathcal{J}(D2GAN)$ **then**
    Encode v by VVC
    flag(v) $\leftarrow$ false
  **else** {flag(previous($v$)) and flag(next($v$))}
    generate $v$ by D2GAN
    flag($v$) $\leftarrow$ true
  **end if**
**end for**

encoder computes the rate distortion (RD) cost function $J$ given by (5.5) for both the VVC decoded view and the one synthesized by the D2GAN.

$$\mathcal{J} = D + \lambda\, R \tag{5.5}$$

where $\lambda$ is the Lagrangian multiplier, $D$ is the distortion and $R$ is the rate in bpp. To set the Lagrangian multiplier ($\lambda$), we empirically determine its value by testing a large set of LF images. We found that the value of 0.1 for $\lambda$ is optimal and for which the Lagrangian optimization is giving the best performance.

At the decoder side, the dropped views are synthesized using D2GAN block. As a reminder, the D2GAN is composed of a generator $G$ and two discriminators $D_1$ and $D_2$. $G$ consists of two CNNs [KWR16], the first CNN estimates the disparity and the second one generates the color image. A set of features (mean and standard deviation) of a sparse set of views (16 views) are fed to the disparity CNN that estimates the disparity at an intermediate view, and then used it to warp (backward) all the input views to the intermediate view. The second color CNN uses all the warped images, derived from the first CNN, to predict the color and synthesizes the dropped views.

Given that the generator $G$ and discriminators ($D1$ and $D2$) are CNN-based blocks, a training phase is required to fix respectively their parameters $\theta_G$, $\theta_{D1}$ and $\theta_{D2}$. Unlike GAN, in D2GAN, the scores returned by $G$ are values in $\mathbb{R}^+$ rather than probabilities in $[0, 1]$. The discriminators and generator are alternatively updated using stochastic

Table 5.1: The average coding gains in terms of BD-BR of D2GAN, trained with reconstructed views, in comparison with the anchor D2GAN training with original views.

| | wPSNR-based | | SSIM-based | |
|---|---|---|---|---|
| | BD-BR | BD-PSNR | BD-BR | BD-SSIM |
| vs. D2GAN Reconstructed | -11.0% | 0.25 | -20.3% | 0.013 |
| vs. D2GAN Recons. separately | **-16.6%** | **0.39** | **-25.5%** | **0.022** |

Table 5.2: BD-BR and BD-PSNR gains calculated against anchor method described in [LWL$^+$16]. 1) Bikes 2) DangerDeMort 3) Flowers 4) Ankylosaurus_Diplodocus 1 5) Aloe 6) Stone_pillars_outside 7) Bedroom 8) Desktop 9) Herbs.

| | VVC-All | | Jia *et al.* [JZW$^+$18] | | Hou *et al.* [HCC19] | | Proposed | |
|---|---|---|---|---|---|---|---|---|
| Im. | BD-BR | BD-PSNR | BD-BR | BD-PSNR | BD-BR | BD-PSNR | BD-BR | BD-PSNR |
| *1)* | –11.7% | 0.72 | –6.3% | 0.48 | –6.9% | 0.49 | **–22.4%** | **0.96** |
| *2)* | –7.8% | 0.22 | –10.8% | 0.28 | –8.7% | 0.26 | **–16.5%** | **0.40** |
| *3)* | –12.3% | 0.56 | –11.9% | 0.54 | –16.2% | 0.72 | **–16.6%** | **0.74** |
| *4)* | –13.2% | 0.44 | –14.9% | –0.72 | –12.3% | 0.39 | **–18.0%** | **0.57** |
| *5)* | –26.4% | 0.85 | –9.1% | 0.31 | –2.46% | –0.12 | **–42.3%** | **1.23** |
| *6)* | –18.3% | 0.61 | –15.1% | 0.52 | –11.9% | 0.28 | **–35.6%** | **0.98** |
| *7)* | –5.3% | 0.46 | –4.0% | 0.32 | –2.3% | 0.18 | **–9.5%** | **0.85** |
| *8)* | –19.6% | 0.32 | –7.5% | 0.11 | 44.1% | –0.61 | **–26.3%** | **0.45** |
| *9)* | –26.0% | 1.14 | –4.4% | –0.11 | 6.9% | –0.20 | **–29.8%** | **1.32** |
| Av. | –15.6% | 0.59 | –8.3% | 0.35 | –0.54% | 0.15 | **–24.1%** | **0.83** |

gradient ascent and descent, respectively. The backward propagation of errors (i.e., cost functions) is applied to update the discriminators and generator with mini-batch size equal to $M$, as shown in Figure 5.2.

For the training phase of D2GAN, 3 configurations were considered : 1) training with the original views , 2) training with reconstructed views at multiple distortion levels including the original views and 3) training for each one distortion level separately. We compared the three configurations, and the obtained results are given in Table 5.1. Based on these results, the third configuration, i.e., D2GAN reconstructed separately, outperforms the other configurations and hence we used it for the D2GAN training.

## 5.4 Results and Discussions

### 5.4.1 Experimental Setup

The proposed deep learning-based architecture described in the previous section was trained with 140 LF images, where 70 LF images are from EPFL dataset [RE16], 50 LF images are from Stanford Lytro LF image dataset [RMS16] and 20 LF images are from HCI dataset [HJKG16]. Each sub-aperture view was splitted into patches of size

60×60, thus resulting in more than 150,000 patches that were used in the training phase. For the testing phase, 9 LF images are selected, 6 LF images are from EPFL dataset [RE16], 1 LF image from Stanford Lytro LF dataset [RMS16] and 2 LF images from HCI dataset [HJKG16], as shown in Figure 5.4. Each of these LF images is composed of 8×8 sub-aperture views. These views are rearranged in a pseudo sequence using spiral order scan and coded using VVC in RA coding configuration at 4 QP values of 18, 24, 28 and 32.

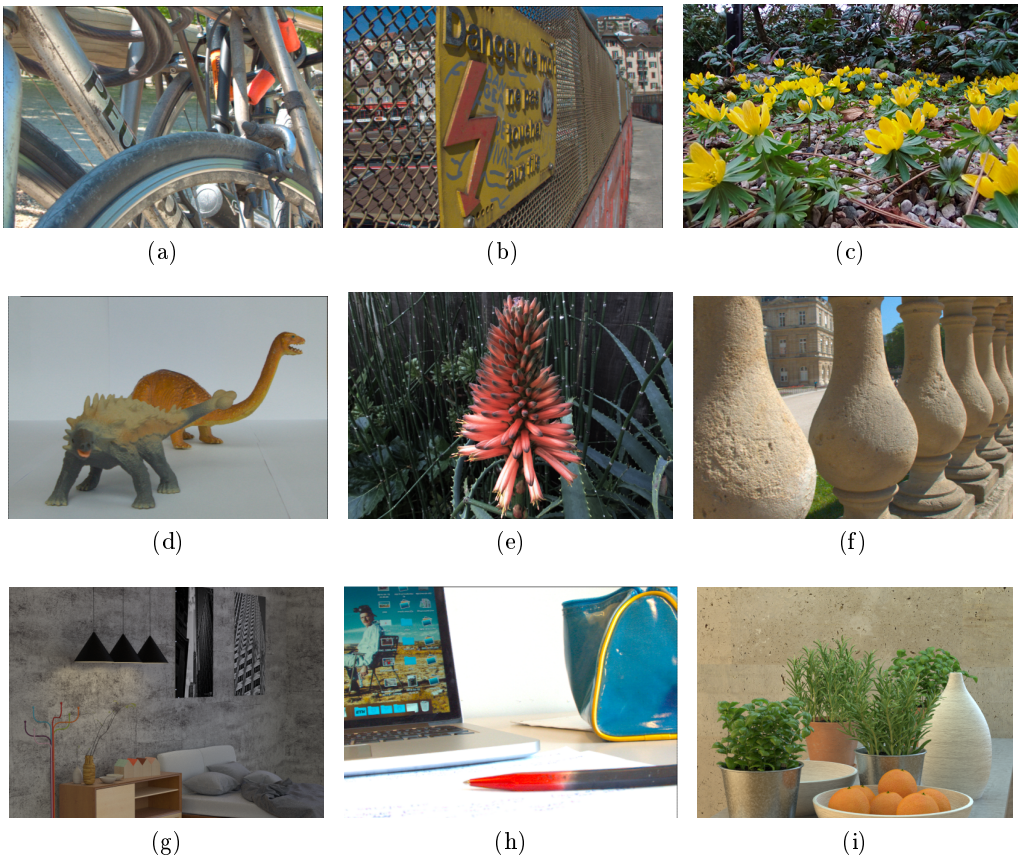



Figure 5.4: Thumbnails of the considered nine LF images: a) Bikes b) DangerDeMort c) Flowers d) Ankylosaurus_Diplodocus 1 e) Aloe f) Stone_pillars_outside g) Bedroom h) Desktop i) Herbs.

The training configuration of D2GAN was set as follows: we trained the generator $G$ and two discriminators ($D_1$ and $D_2$) with the ADAM optimizer [KB14] by setting $\beta_1 = 0.9$, $\beta_2 = 0.999$, learning rate= 0.0002, batch-size= 10 and kernel size of convolutional layers as depicted in Figure 5.2. The regularization coefficients of $D_1$ and $D_2$ was set as $\alpha = 0.2$ and $\beta = 0.2$, respectively. For the generator, we used input patch of 60×60, stride= 16, and output patch= $36 \times 36$ (reduced size is due to the convolutions).

### 5.4.2 Evaluations

We compared the proposed scheme with four state-of-the-art methods: 1) VVC-All that encodes all views with the VVC in RA coding configuration, 2) LF-GAN method proposed in [JZW+18], where a sub-set of views are coded with HEVC, while the remaining views are generated by GAN and the residual error of views are transmitted to the decoder, 3) the method proposed in [LWL+16] encoding the views as a pseudo-video sequence using specific order scan, 4) the method of Hou el al. [HCC19] that exploits the inter- and intra-views correlation to encode the views using HEVC. The latter method is considered as the anchor method.

### 5.4.3 Results

The BD-BR [Bjø01, hB08] is a PSNR based metric. It is used in this chapter to assess the gain of the proposed approach compared to the anchor solution. A negative BD-BR value refers to a bitrate reduction compared to the anchor method, while a positive value expresses a bitrate overhead.

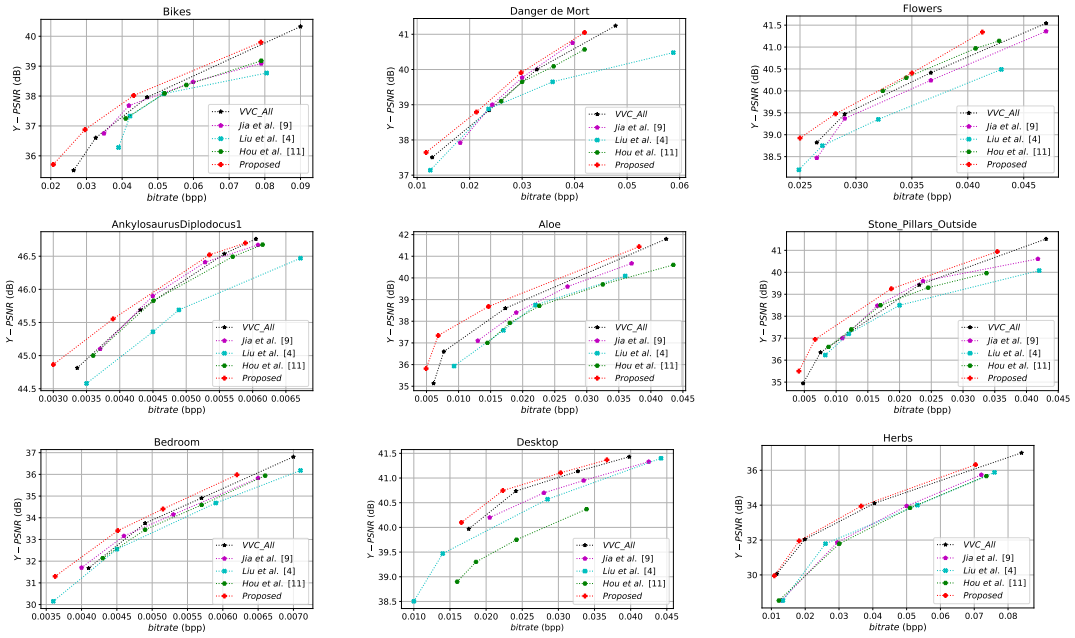



Figure 5.5: RD curves of the five considered solutions for the 9 LF images using four QP values.

R-D curves based on PSNR for the 9 LF images are provided in Figure 5.5. We can notice that for all considered images, the proposed coding method provides the highest performance for all bitrates. The previous conclusion is confirmed by Table 5.2, providing the Bjøntegaard results of the four coding solutions compared to the anchor one [LWL+16]. The proposed method achieved an average BD-BR gain of -24.1% and

Table 5.3: Running time in seconds of the four LF image coding methods.

| QP | Encoder | | | | |
|---|---|---|---|---|---|
| | VVC-All | Jia *et al.* [JZW+18] | Hou *et al.* [HCC19] | Our | |
| | CPU | GPU | CPU | CPU | GPU |
| 18 | **259** | 450 | 6028 | 559 | 449 |
| 22 | **152** | 350 | 6028 | 452 | 342 |
| 28 | **101** | 220 | 6028 | 401 | 291 |
| 34 | **66** | 142 | 6028 | 366 | 256 |
| Average | **66** | 291 | 6028 | 445 | 335 |
| | Decoder | | | | |
| Average | **4** | 53 | 583 | 124 | 94 |

BD-PSNR of 0.83 dB compared to the anchor method [LWL+16].

The complexity of the proposed coding approach is also evaluated and compared to the other methods on both CPU and GPU platforms. The performance has been carried-out on an Intel core i9-7900X CPU running at 3.3GHz PC with 64 GB memory and a TITAN Xp NVDIA GPU. It is important to note that the GPU is only used when the D2GAN block is involved in the coding scheme.

Table 5.3 gives the encoding and decoding run times in seconds. We can notice that the proposed solution requires almost the same complexity in the encoding for all QP compared to [JZW+18] and [HCC19] methods. The GPU enables to speedup the encoding part related to the D2GAN block. However, the decoder of the proposed solution is more complex than the other solutions due the D2GAN block.

## 5.5 Conclusion

In this chapter, we have proposed a view synthesis based LF image compression approach. In the proposed coding scheme, a set of views are encoded using VVC, while the remaining views are dropped. The dropped views are synthesized using enhanced GAN-based approach known as D2GAN. The transmitted and dropped views are selected using RDO process. In addition, in order to avoid impacting the decoder with the dropped views, the latter are determined according to the temporal scalability of VVC. All these features allow reducing bitrate required by LF image, while providing views with high visual quality.

The experiments results show the efficiency of our scheme, which achieved bitrate reduction of $-24.1\%$ in terms of BD-BR and increased the visual quality by 0.83 dB in BD-PSNR.

# Chapter 6

# Conclusion and Perspectives

## General Conclusion

In this thesis, several contributions have been proposed. The conducted works have been done with the aim to develop methods for efficient LF image coding based on standard 2D encoders and Deep Learning techniques while providing optimal quality of experience.
Below, we summarize the contributions of the thesis, and then propose some directions for future research.

Firstly, we introduced a LF coding solution based on the linear approximation and CNN, where only a small set of views in a LF image is coded. The dropped views are either linearly approximated or generated by the trained CNN based on a RDO scheme. The training of CNN is applied by providing a sequence of random mini-batches of LF images uniformly selected from the entire training LF dataset. The value of the Lagrangian multiplier in the RDO scheme was empirically set after an exhaustive testing over a large set of LF images.

As a second contribution, we conducted a subjective test for visual quality assessment of LF contents, using a framework recording user interaction and analyzing how Quality of Experience (QoE) is affected by compression distortions. Two recent LF compression methods based on view synthesis have been compared subjectively and objectively to two pseudo-video sequences based coding approaches. Experimental results show that the method based on view synthesis achieves significant better coding performance without affecting the visual quality. Specifically, the subjective quality assessment showed that the view synthesis based method provides substantial superior visual quality, especially at low and medium bitrates.

Finally, the last contribution following the same way as the first one contribution, consisting of encoding a sparse set of views, and estimating the rest of views at the decoder side. In particular, the first set of selected reference views are coded with the next generation video coding standard called VVC, while the second set of views are either synthesized from the first decoded set of views using a D2GAN or decoded by a VVC decoder. The D2GAN has been trained with a large set of LF images coded at

different distortions. The architecture offered by the D2GAN, composed by a generator and two discriminators, enables better training and thus synthesizes views with highly visual quality. In addition, to increase the coding efficiency, a RDO is adopted to select which views should be encoded and transmitted and which ones should be dropped and synthesized at the decoder side.

## Future Work and Perspectives

As future work, we can propose a method to determine the optimal value for higher efficiency of the RDO scheme. Moreover, we can perform the coding step by using Densely Connected Convolutional Networks (DenseNet) [HLvdMW17]. The DenseNet connects each layer to every other layer in a feed-forward fashion, whereas traditional convolutional networks with $L$ layers have $L$ connections - one between each layer and its subsequent layer. For each layer, the feature-maps of all preceding layers are used as inputs, and its own feature-maps are used as inputs into all subsequent layers.

In addition, it could be interesting to measure whether another factor as refocusing views has any impact on observer's voting. Future works can include more LF compression methods based on view synthesis, as well as other more recent compression methods for LF images.

Finally, using Curriculum Learning (CL) in the training phase can improve the performance of LF compression methods based on view synthesis. The idea of human CL attempts to impose a structure on the training group [BLCW09]. Such a structure is essentially based on a notion of "easy" and "difficult" examples and uses this distinction in order to teach the learner to generalize easier examples before more difficult examples. Thus, we can classify the LF images used during the training stage according to their complexity in relation to their contents. For instance, images with lot of details, i.e., highly textured, could be considered as difficult images and vice versa.

# Author's publications

N. Bakir, W. Hamidouche, O. Deforges, K. Samrouth and M. Khalil, "Light field image compression based on convolutional neural networks and linear approximation," in Proc. IEEE International Conference on Image Processing (ICIP), Oct 2018, pp. 1128-1132.

N. Bakir, W. Hamidouche, K. Samrouth, S. FEZZA, O. Deforges and M. Khalil, "RDO-based Light Field Image Coding using Convolutional Neural Networks and Linear Approximation", IEEE Data Compression Conference (DCC), Snowbird, UT, USA, 2019, pp. 554-554.

N. Bakir, S. Fezza, W. Hamidouche, K. Samrouth and O. Deforges, "Subjective Evaluation Of Light Field Image Compression Methods Based On View Synthesis",IEEE 27th European Signal Processing Conference (EUSIPCO), Coruna, Spain, Sep 2019.

N. Bakir, W. Hamidouche, S. Fezza, K. Samrouth, O. Deforges and M. Khalil, "Light field image coding using dual discriminator generative adversarial network and VVC temporal scalability", IEEE International Conference on Multimedia and Expo, (ICME), London, UK, July, 2020.

N. Bakir, W. Hamidouche, S. Fezza, K. Samrouth, O. Deforges and M. Khalil, "Light Field Image Coding Using VVC standard and View Synthesis based on Dual Discriminator GAN", IEEE Transactions on Multimedia, USA, December, 2020.

# Glossary

AI : All Intra
AVC : Advanced Video Coding
BD-BR : Bjøntegaard Delta Bit Rate
BD-PS : Bjøntegaard Delta PSNR
BDR : Bjøntegaard Delta Bit Rate
bpp : bits per pixel
CABAC : Context-Adaptive Binary Arithmetic Coding
CALIC : Context-based Adaptive lossless Image Codec
CL : Curriculum Learning
CNN : Convolutional Neural Networks
CTU : Coding Tree Unit
CU : Coding Unit
CV : Central View
D2GAN : Dual Discriminator Generative Adversarial Nets
DenseNet : Densely Connected Convolutional Networks
DCT : Discrete Cosine Transform
DL : Deep Learning
DoF : Depth of Field
DSIS : Double Stimulus Impairment Scale
DSCQS : Double Stimuli Continuous Quality Scale
DWT : Discrete Wavelet Transform
EPI : Epipolar Plane Image
fps : frames per second
GAN : Generative Adversarial Network
GOP : Group of Pictures
GOV : Groups of Views
H2DC-CNN : Hybrid 2D video codec CNN
HEVC : High Efficiency Video Coding
HLRA : Homography Low Rank Approximation
HM HEVC : Reference Model
HMD : Head Mounted Displays
HSP2P : Hierarchical Superpixel-to-Pixel Dense Image Matching
ITU : International Telecommunication Union
JCT-VC : Joint Collaborative Team on Video Coding

JEM : Joint Exploration Model
JVET : Joint Video Exploration Team
LA : Linear Approximation
LDP : Low Delay P
LF : Light Field
MOS : Mean Opinion Score
MPEG : Motion Picture Expert Group
MSE : Mean Squared Error
MV : Motion Vector
MV-HEVC : Multi-View extension of High Efficiency Video Coding
NN : Neural Network
POC : Picture Order Count
PSNR : Peak Signal to Noise Ratio
QoE : Quality of Experience
QP : Quantization Parameter
RA : Random Access
RD : Rate Distortion
RDO : Rate Distortion Optimization
PU : Prediction Unit
SIFT : Scale-Invariant Feature Transform
simulcast : Simultaneous Broadcast
SL : Single Layer
SLIC : Simple Linear Iterative Clustering
SPGL1 : Spectral Projected Gradient for L1
SSCQS : Single Stimulus Continuous Quality Scale
SSIM : Structural Similarity
TU : Transform Unit
VR/AR : Virtual Reality, Augmented Reality
VVC : Versatile Video Coding

# Bibliography


[AB91] Edward H. Adelson and James R. Bergen. The plenoptic function and the elements of early vision. In Computational Models of Visual Processing, pages 3–20. MIT Press, 1991.

[Ada02] A. Adams. Stanford (new) light field archive. In http://lightfield.stanford.edu/lfs.html, Stanford Graphics Laboratory, 2002.

[Agg11] A. Aggoun. Compression of 3d integral images using 3d wavelet transform. Journal of Display Technology, 7(11):586–592, Nov 2011.

[AOS17] W. Ahmad, R. Olsson, and M. Sjostrom. Interpreting plenoptic images as multi-view sequences for improved compression. In 2017 IEEE International Conference on Image Processing (ICIP), pages 4557–4561, Sep. 2017.

[APKS18] W. Ahmad, L. Palmieri, R. Koch, and M. Sjostrom. Matching light field datasets from plenoptic cameras 1.0 and 2.0. In 2018 - 3DTV-Conference: The True Vision - Capture, Transmission and Display of 3D Video (3DTV-CON), pages 1–4, June 2018.

[APP18] H. Amirpour, M. Pereira, and A. Pinheiro. High efficient snake order pseudo-sequence based light field image compression. In 2018 Data Compression Conference, pages 397–397, March 2018.

[APP+19] H. Amirpour, A. Pinheiro, M. Pereira, F. Lopes, and M. Ghanbari. Light field image compression with random access. In 2019 Data Compression Conference (DCC), pages 553–553, March 2019.

[ASS+12] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk. Slic superpixels compared to state-of-the-art superpixel methods. IEEE Transactions on Pattern Analysis and Machine Intelligence, 34(11):2274–2282, Nov 2012.

[BCC18] F. Battisti, M. Carli, and P. L. Callet. A study on the impact of visualization techniques on light field perception. In 2018 26th European Signal Processing Conference (EUSIPCO), pages 2155–2159, Sep. 2018.

[Beg18] Jean Begaint. Towards novel inter-prediction methods for image and video compression. INRIA Rennes - Bretagne Atlantique et Technicolor R.I. PhD thesis, 11 2018.

[BHD+18] N. Bakir, W. Hamidouche, O. Déforges, K. Samrouth, and M. Khalil. Light field image compression based on convolutional neural networks and linear approximation. In 2018 25th IEEE International Conference on Image Processing (ICIP), pages 1128–1132, Oct 2018.

[Bjø01] G. Bjøntegaard. Calcuation of average psnr differences between rd-curves. VCEG-M33 ITU-T Q6/16, Austin, TX, USA, 2-4 April, 2001.

[BLCW09] Y. Bengio, J. Louradour, R. Collobert, and J. Weston. Curriculum learning. In Proceedings of the 26th Annual International Conference on Machine Learning, ICML 09, page 41, USA, 2009. Association for Computing Machinery.

[BT.12a] ITU-R BT.2022. General viewing conditions for subjective assessment of quality of sdtv and hdtv television pictures on flat panel displays. In International Telecommunication Union, Aug. 2012.

[BT.12b] ITU-R BT.500-13. Methodology for the subjective assessment of the quality of television pictures. In International Telecommunication Union, Jan. 2012.

[BZDG18] Y. Bai, Y. Zhang, M. Ding, and B. Ghanem. Sod-mtgan: Small object detection via multi-task generative adversarial network. In The European Conference on Computer Vision (ECCV), September 2018.

[CC15] J.-S. Chen and D. P. Chu. Improved layer-based method for rapid hologram generation and real-time interactive holographic display applications. Opt. Express, 23(14):18143–18155, Jul 2015.

[CHB17] L. Cavigelli, P. Hager, and L. Benini. Cas-cnn: A deep convolutional neural network for image compression artifact suppression. In 2017 International Joint Conference on Neural Networks (IJCNN), pages 752–759, May 2017.

[Dan14] D. G. Dansereau. Plenoptic signal processing for robust vision in field robotics. Ph.D. dissertation, University of Sydney Graduate School of Engineering and IT School of Aerospace, Mechanical and Mechatronic Engineering, 2014.

[DBPW13] D. Dansereau, D. Bongiorno, O. Pizarro, and S. Williams. Light field image denoising using a linear 4d frequency-hyperfan all-in-focus filter. volume 8657, 01 2013.

[DBPW19] D. Dansereau, D. Bongiorno, O. Pizarro, and S. Williams. Light field display technical deep dive, fovi 3d. 2019.

[DH03] S. E. Suesstrunk D. Hasler. Measuring colorfulness in natural images, 2003.

[DLJG19] E. Dib, M. Le Pendu, X. Jiang, and C. Guillemot. Super-ray based low rank approximation for light field compression. In 2019 Data Compression Conference (DCC), pages 369–378, March 2019.

[DPG19] E. Dib, M. L. Pendu, and C. Guillemot. Light field compression using fourier disparity layers. In 2019 IEEE International Conference on Image Processing (ICIP), pages 3751–3755, Sep. 2019.

[DPW13] D. G. Dansereau, O. Pizarro, and S. B. Williams. Decoding, calibration and rectification for lenselet-based plenoptic cameras. In 2013 IEEE Conference on Computer Vision and Pattern Recognition, pages 1027–1034, June 2013.

[DQW04] Xu Dong, Dai Qionghan, and Xu Wenli. Data compression of light field using wavelet packet. In 2004 IEEE International Conference on Multimedia and Expo (ICME) (IEEE Cat. No.04TH8763), volume 2, pages 1071–1074 Vol.2, June 2004.

[DSS17] X. Dong, J. Shen, and L. Shao. Hierarchical superpixel-to-pixel dense matching. IEEE Transactions on Circuits and Systems for Video Technology, 27(12):2518–2526, Dec 2017.

[EFPS16] T. Ebrahimi, S. Foessel, F. Pereira, and P. Schelkens. Jpeg pleno: Toward an efficient representation of visual reality. IEEE MultiMedia, 23(4):14–20, Oct 2016.

[EPdRH02] M. Egmont-Petersen, D. de Ridder, and H. Handels. Image processing with neural networks, a review. Pattern Recognition, 35(10):2279 – 2301, 2002.

[Ga14] I. Goodfellow and et. all. Generative adversarial nets. In Advances in neural information processing systems, pages 2672–2680, 2014.

[GGSC96] Steven J. Gortler, Radek Grzeszczuk, Richard Szeliski, and Michael F. Cohen. The lumigraph. In Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH '96, pages 43–54, New York, NY, USA, 1996. ACM.

[GJK+17] M. Gupta, A. Jauhari, K. Kulkarni, S. Jayasuriya, A. Molnar, and P. Turaga. Compressive light field reconstructions using deep learning. In 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pages 1277–1286, July 2017.

[HART17] P. Helin, P. Astola, B. Rao, and I. Tabus. Minimum description length sparse modeling and region merging for lossless plenoptic image compression. IEEE Journal of Selected Topics in Signal Processing, 11(7):1146–1161, Oct 2017.

[HAS+18] X. Huang, P. An, L. Shan, R. Ma, and L. Shen. View synthesis for light field coding using depth estimation. In 2018 IEEE International Conference on Multimedia and Expo (ICME), pages 1–6, July 2018.

[HASM18] X. Huang, P. An, L. Shen, and R. Ma. Efficient light field images compression method based on depth estimation and optimization. IEEE Access, 6:48984–48993, 2018.

[hB08] G. Bjøntegaard. Improvements of the bd-psnr model. ITU-T SG16 Q, 6:35, 2008.

[HCC19] J. Hou, J. Chen, and L. Chau. Light field image compression based on bi-level view compensation with rate-distortion optimization. IEEE Trans. Cir. and Sys. for Video Technol., 29(2):517–530, February 2019.

[HJKG16] K. Honauer, O. Johannsen, D. Kondermann, and B. Goldlücke. A dataset and evaluation methodology for depth estimation on 4d light fields. In Shang-Hong Lai, editor, Computer Vision - ACCV 2016 : 13th Asian Conference on Computer Vision, Taipei, Taiwan, November 20-24, 2016, Revised Selected Papers, Part III, Cham, 2016. Springer, Springer.

[HLvdMW17] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger. Densely connected convolutional networks. CVPR, 2017.

[HSG17] M. Hog, N. Sabater, and C. Guillemot. Superrays for efficient light field processing. IEEE Journal of Selected Topics in Signal Processing, 11(7):1187–1199, Oct 2017.

[HZ10] A. Hore and D. Ziou. Image quality metrics: Psnr vs. ssim. In 2010 20th International Conference on Pattern Recognition, pages 2366–2369, Aug 2010.

[ITU] User requirements for objective perceptual video quality measurements in digital cable television.

[Jia99] J. Jiang. Image compression with neural networks, a survey. Signal Processing: Image Communication, 14(9):737 – 760, 1999.

[JLG17] X. Jiang, M. Le Pendu, and C. Guillemot. Light field compression using depth image based view synthesis. In 2017 IEEE International Conference on Multimedia Expo Workshops (ICMEW), pages 19–24, July 2017.

[jpe18] "iso/iec jtc 1/sc29/wg1 jpeg". Jpeg pleno light field coding vm 1.1, 2018.

[JPF+17] X. Jiang, M. Le Pendu, R. A. Farrugia, S. S. Hemami, and C. Guillemot. Homography-based low rank approximation of light fields for compression. In 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 1313–1317, March 2017.

[JPFG17] X. Jiang, M. Le Pendu, R. A. Farrugia, and C. Guillemot. Light field compression with homography-based low-rank approximation. IEEE Journal of Selected Topics in Signal Processing, 11(7):1132–1145, Oct 2017.

[JZW+18] C. Jia, X. Zhang, S. Wang, S. Wang, S. Pu, and S. Ma. Light field image compression using generative adversarial network based view synthesis. IEEE Journal on Emerging and Selected Topics in Circuits and Systems, pages 1–1, 2018.

[KB14] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. CoRR, abs/1412.6980, 2014.

[KTF18] K. Komatsu, K. Takahashi, and T. Fujii. Scalable light field coding using weighted binary images. In 2018 25th IEEE International Conference on Image Processing (ICIP), pages 903–907, Oct 2018.

[KWR16] N. Khademi Kalantari, T.C. Wang, and R. Ramamoorthi. Learning-based view synthesis for light field cameras. ACM Trans. Graph., 35(6):193:1–193:10, November 2016.

[KY18] K. Wakunami L. Jorissen Y. Ichihashi M. Okui R. Oi Kenji Y., B. J. Jackin. Hologram printing for next-generation holographic display, 2018.

[LCCL08] O. Lézoray, C. Charrier, H. Cardot, and S. Lefèvre. Machine learning in image processing. EURASIP Journal on Advances in Signal Processing, 2008(1):927950, May 2008.

[LG09] A. Lumsdaine and T. Georgiev. The focused plenoptic camera. In 2009 IEEE International Conference on Computational Photography (ICCP), pages 1–8, April 2009.

[LH96] M. Levoy and P. Hanrahan. Light field rendering. In Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH '96, pages 31–42, New York, NY, USA, 1996. ACM.

[LLL+17] L. Li, Z. Li, B. Li, D. Liu, and H. Li. Pseudo sequence based 2-d hierarchical coding structure for light-field image compression. In 2017 Data Compression Conference (DCC), pages 131–140, April 2017.

[Low04] David G. Lowe. Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision, 60(2):91–110, Nov 2004.

[LPT16] Lilin Liu, Zhiyong Pang, and Dongdong Teng. Super multi-view three-dimensional display technique for portable devices. Opt. Express, 24(5):4421–4430, Mar 2016.

[LSOJ14] Y. Li, M. Sjostrom, R. Olsson, and U. Jennehag. Efficient intra prediction scheme for light field image compression. In 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 539–543, May 2014.

[LTH+17] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, and W. Shi. Photo-realistic single image super-resolution using a generative adversarial network. In 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 105–114, July 2017.

[LWL+16] D. Liu, L. Wang, L. Li, Zhiwei Xiong, Feng Wu, and Wenjun Zeng. Pseudo-sequence-based light field image compression. In 2016 IEEE International Conference on Multimedia Expo Workshops (ICMEW), pages 1–4, July 2016.

[LYT+17] Z. Liu, R. A. Yeh, X. Tang, Y. Liu, and A. Agarwala. Video frame synthesis using deep voxel flow. In 2017 IEEE International Conference on Computer Vision (ICCV), pages 4473–4481, Oct 2017.

[LZM09] M. LEVOY, Z. ZHANG, and I. MCDOWALL. Recording and controlling the 4d light field in a microscope using microlens arrays. Journal of Microscopy, 235(2):144–162, 2009.

[MWS17] J. Boyce A. Segall M. Wien, V. Baroncini and T. Suzuki. Preliminary joint call for evidence on video compression with capability beyond hevc. Janvier 2017.

[NLB+05] R. Ng, M. Levoy, M. Bredif, G. Duval, M. Horowitz, and P. Hanrahan. Light field photography with a hand-held plenoptic camera. In in Stanford University Computer Science Tech Report, 2005.

[NLVP17] T. Nguyen, T. Le, H. Vu, and D. Phung. Dual discriminator generative adversarial nets. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, Advances in Neural Information Processing Systems 30, pages 2670–2680. Curran Associates, Inc., 2017.

[OSS+12] J. Ohm, G. J. Sullivan, H. Schwarz, T. K. Tan, and T. Wiegand. Comparison of the coding efficiency of video coding standards including high

efficiency video coding. IEEE Transactions on Circuits and Systems for Video Technology, 22(12):1669–1684, Dec 2012.

[P.908] ITU-T P.910. Subjective video quality assessment methods for multimedia applications. In International Telecommunication Union, Apr. 2008.

[PA16] C. Perra and P. Assuncao. High efficiency coding of light field images based on tiling and pseudo-temporal data arrangement. In 2016 IEEE International Conference on Multimedia Expo Workshops (ICMEW), pages 1–4, July 2016.

[PdSL+17] F. Pereira, E. A. B. da Silva, G. Lafruit, R. Chellappa, and S. Theodoridis. Plenoptic imaging: Representation and processing. 2017.

[Per15] C. Perra. Lossless plenoptic image compression using adaptive block differential prediction. In 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 1231–1234, April 2015.

[PGL+17] P. Paudyal, J. Gutiérrez, P. Le Callet, M. Carli, and F. Battisti. Characterization and selection of light field content for perceptual assessment. In 2017 Ninth International Conference on Quality of Multimedia Experience (QoMEX), pages 1–6, May 2017.

[PMG+17] A. Prakash, N. Moran, S. Garber, A. Dilillo, and J. Storer. Semantic perceptual image compression using deep convolution networks. In 2017 Data Compression Conference (DCC), pages 250–259, April 2017.

[Pom91] D. A. Pomerleau. Efficient training of artificial neural networks for autonomous navigation. Neural Computation, 3(1):88–97, March 1991.

[PP93] N. R Pal and S. K Pal. A review on image segmentation techniques. Pattern Recognition, 26(9):1277 – 1294, 1993.

[RAY] Raytrix | 3d light field camera technology, 2018.

[RE16] Martin Rerabek and Touradj Ebrahimi. New light field image dataset. In https://mmspg.epfl.ch/EPFL-light-field-image-dataset, 2016.

[RHPD19] K. Reuze, W. Hamidouche, P. Philippe, and O. Deforges. Dynamic lists for efficient coding of intra prediction modes in the future video coding standard. In 2019 Data Compression Conference (DCC), pages 601–601, March 2019.

[RMS16] S. Raj, L. Michael, and A. Sunder. Stanford lytro light field archive. In http://lightfields.stanford.edu/, 2016.

[RSMG18] M. Rizkallah, X. Su, T. Maugey, and C. Guillemot. Graph-based transforms for predictive light field compression based on super-pixels. In 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 1718–1722, April 2018.

[SGGH17] I. Schiopu, M. Gabbouj, A. Gotchev, and M. M. Hannuksela. Lossless compression of subaperture images using context modeling. In 2017 3DTV Conference: The True Vision - Capture, Transmission and Display of 3D Video (3DTV-CON), pages 1–4, June 2017.

[SHD+] N. Sidaty, W. Hamidouche, O. Deforges, P. Philippe, and J. Fournier. Compression performance of the versatile video coding: Hd and uhd visual quality monitoring.

[SHD+19] N. Sidaty, W. Hamidouche, O. Déforges, P. Philippe, and J. Fournier. Compression Performance of the Versatile Video Coding: HD and UHD Visual Quality Monitoring. In Picture Coding Symposium (PCS), November 2019.

[SHDP17] N. Sidaty, W. Hamidouche, O. Deforges, and P. Philippe. Compression Efficiency of the Emerging Video Coding Tools. In IEEE Conference on Image Processing (ICIP), September 2017.

[SM18] I. Schiopu and A. Munteanu. Macro-pixel prediction based on convolutional neural networks for lossless compression of light field images. In 2018 25th IEEE International Conference on Image Processing (ICIP), pages 445–449, Oct 2018.

[SOHW12] G. J. Sullivan, J. Ohm, W. Han, and T. Wiegand. Overview of the high efficiency video coding (hevc) standard. IEEE Transactions on Circuits and Systems for Video Technology, 22(12):1649–1668, Dec 2012.

[SZ17] Z. Shengyang and C. Zhibo. Light field image coding via linear approximation prior. In IEEE International Conference on Image Processing, China, January 2017.

[TZAG13] G. Todor, Y. Zhan, L. Andrew, and Sergio G. Lytro camera technology: theory, algorithms, performance analysis, 2013.

[VrE17] I. Viola, M. rerabek, and T. Ebrahimi. Comparison and evaluation of light field image coding approaches. IEEE Journal of Selected Topics in Signal Processing, 11(7):1092–1106, Oct 2017.

[Wei12] E. W Weisstein. L1-norm. http://mathworld.wolfram.com/l1-norm.html. pages 48, 73, 100, 103, 107, 2012.

[WER16] T. Wang, A. A. Efros, and R. Ramamoorthi. Depth estimation with occlusion modeling using light-field cameras. IEEE Transactions on Pattern Analysis and Machine Intelligence, 38(11):2170–2181, Nov 2016.

[WJV+05] B. Wilburn, N. Joshi, V. Vaish, E.-V. Talvala, E. R. Antúnez, A. Barth, A. Adams, M. Horowitz, and M. Levoy. High performance imaging using large camera arrays. ACM Trans. Graph., 24:765–776, 2005.

[WSBL03] T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra. Overview of the h.264/avc video coding standard. IEEE Transactions on Circuits and Systems for Video Technology, 13(7):560–576, July 2003.

[WZW+17] G. Wu, M. Zhao, L. Wang, Q. Dai, T. Chai, and Y. Liu. Light field reconstruction using deep convolutional network on epi. In 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 1638–1646, July 2017.

[YEBM02] J. C. Yang, M. Everett, C. Buehler, and L. McMillan. A real-time distributed light field camera. In Proceedings of the 13th Eurographics Workshop on Rendering, EGRW '02, pages 77–86, Aire-la-Ville, Switzerland, Switzerland, 2002. Eurographics Association.

[YLX16] L. Yao, Y. Liu, and W. Xu. Real-time virtual view synthesis using light field. EURASIP Journal on Image and Video Processing, 2016(1):25, Sep 2016.

[ZBSS04] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: from error visibility to structural similarity. IEEE Transactions on Image Processing, 13(4):600–612, April 2004.

[ZC17] S. Zhao and Z. Chen. Light field image coding via linear approximation prior. In 2017 IEEE International Conference on Image Processing (ICIP), pages 4562–4566, Sep. 2017.

[ZCYH16] S. Zhao, Z. Chen, K. Yang, and H. Huang. Light field image coding with hybrid scan order. In 2016 Visual Communications and Image Processing (VCIP), pages 1–4, Nov 2016.

[ZWJ+18] Z. Zhao, S. Wang, C. Jia, X. Zhang, S. Ma, and J. Yang. Light field image compression based on deep learning. In 2018 IEEE International Conference on Multimedia and Expo (ICME), pages 1–6, July 2018.

# List of Figures

# List of Tables

# List of Algorithms

INSA de RENNES
Service des Formations

# AVIS DU JURY SUR LA REPRODUCTION DE LA THESE SOUTENUE

**Titre de la thèse:**
Compression d'images et de vidéos Ligth Field

**Nom Prénom de l'auteur : BAKIR NADER**

Membres du jury :
- Monsieur EL HASSAN Bachar
- Madame SAMROUTH Khouloud
- Monsieur KHALIL Mohamad
- Monsieur DEFORGES Olivier
- Monsieur CAGNAZZO Marco
- Monsieur DUFAUX Frédéric

Président du jury : EL HASSAN Bachar

Date de la soutenance : 10 Juin 2020

Reproduction de la these soutenue

[X] Thèse pouvant être reproduite en l'état
[ ] Thèse pouvant être reproduite après corrections suggérées

Fait à Beyrouth, le 10 Juin 2020

Signature du président de jury

Bachar El Hassan

B. Hassan

Le Directeur,

M'hamed DRISSI

INSTITUT NATIONAL des SCIENCES APPLIQUÉES · RENNES ·

UNIVERSITE BRETAGNE LOIRE / MATHSTIC

INSA RENNES

## Titre : Compression d'images et de vidéos Light Field

**Mots clés :** Champs Lumineux, Apprentissage Profond, Futur Codage Vidéo, Evaluation Subjective

**Résumé :** Les applications de vision par ordinateur telles que le refocusing, la segmentation et la classification deviennent l'un des services les plus avancés dans le domaine de traitement d'image mais de telles applications nécessitent des informations sémantiques riches de la scène. La technologie 3D est largement utilisée dans les domaines de divertissement, d'imagerie médicale et de l'éducation. Il existe différentes manières de représenter l'information 3D. Une technologie récente dont l'importance est grandissante est proposée par les images Light Field (LF). L'image LF est une image non conventionnelle contenant des informations denses telles que l'intensité des rayons lumineux qui interagissent avec la scène. Cependant, un tel système d'imagerie présente de nombreux inconvénients, notamment une grande quantité de données produites. Des techniques de compression adaptées sont ainsi nécessaires. L'objectif de cette thèse est donc de développer des méthodes efficaces pour la compression d'images et de vidéos Light Field.

Le succès récent de l'apprentissage profond dans divers domaines notamment dans les domaines du traitement des images et du son, a été établi comme un facteur clé dans nos travaux de recherches. La première partie de cette thèse propose un schéma de codage du champ lumineux basé sur CNN qui inclut RDO suivi d'un post-traitement. Le concept principal est d'exploiter la corrélation entre les différentes vues LF et d'éviter le codage de toutes les vues. Ainsi, un ensemble de vues LF est codé par un codeur 2D standard, puis les autres sont soit estimées par une approximation linéaire soit générées par CNN. Dans un second temps, une comparaison subjective entre les solutions de codage proposées et les standards ont montré des gains très significatifs. Enfin, la dernière partie de cette thèse a consisté à intégrer un Dual Discriminative Generative Adverserial Network (D2GAN) dans l'encodeur standard hiérarchique Versatile Video Coding (VVC). L'idée globale est de coder les vues du niveau hiérarchique supérieur et les générer avec D2GAN au niveau du décodeur.

## Title : Light Field Image and Video Compression

**Keywords :** Light Field, Deep Learning, Future Video Coding, Subjective Evaluation

**Abstract:** Computer vision applications such as refocusing, segmentation and classification are becoming one of the most advanced services in the field of image processing, but such applications require rich semantic information of the scene. 3D technology is widely used in the fields of entertainment, medical imaging and education. There are different ways of representing 3D information. A recent technology of growing importance is Light Field (LF) images. The LF image is an unconventional image containing dense information such as the intensity of the light rays interacting with the scene. However, such an imaging system has many drawbacks, including the large amount of data produced. This requires appropriate compression techniques. The goal of this thesis is to develop new methods for efficient LF image and video compression. The recent success in deep learning in various fields, particularly in the areas of image and speech processing. Thus, the overall established as a key factor in this research work. The first part of this thesis proposes a CNN-based light field coding scheme that includes RDO followed by post-processing. The main concept is to exploit the correlation between the different LF views and avoid coding of all views. So, one set of LF views is coded by a standard 2D encoder while others are estimated by linear approximation or generated by CNN.
The second part shows very significant gains while drawing a subjective comparison between the proposed coding solutions and the standards. Finally, the last part of this thesis consists in integrating a Dual Discriminative Generative Adverserial Network (D2GAN) into the standard hierarchical Versatile Video Coding (VVC) encoder. The overall idea is to encode the views of the upper hierarchical level and generate them with D2GAN at the decoder side.