Deep Learning for Near-duplicated Patterns Discovery and Alignment in Artworks

Xi Shen

École Doctorale MSTIC
Mathématiques & Sciences et Technologies
de l'Information et de la Communication

Thèse de doctorat
de École des Ponts

Domaine : Traitement du Signal et des Images

Présentée par
Xi Shen

pour obtenir le grade de

Docteur de École des Ponts

Deep Learning for Near-duplicated Patterns Discovery and Alignment in Artworks

---

Soutenue publiquement le 3 Decembre 2021 devant le jury composé de :

Yann GOUSSEAU	Professor, Telecom ParisTech	Rapporteur
Laurent HEUTTE	Professor, Université de Rouen Normandie	Rapporteur
Patrick PEREZ	Director of Research, Valeo AI	Examinateur
Alexei A. EFROS	Professor, UC Berkeley	Examinateur
Shiry GINOSAR	Postdoctoral Researcher, UC Berkeley	Examinatrice
Aubry MATHIEU	Senior Researcher, École des Ponts	Directeur de thèse

École des Ponts ParisTech
LIGM-IMAGINE
6, Av Blaise Pascal - Cité Descartes
Champs-sur-Marne
77455 Marne-la-Vallée cedex 2
France

Université Paris-Est Marne-la-Vallée
École Doctorale Paris-Est MSTIC
Département Études Doctorales
6, Av Blaise Pascal - Cité Descartes
Champs-sur-Marne
77454 Marne-la-Vallée cedex 2
France

This manuscript is dedidated to my grandfather.
仅以此文献给我的祖父。

Abstract

The goal of this thesis is to develop deep learning methods for artwork analysis. More precisely, we focus on three particular tasks: style-invariant patterns retrieval, near-duplicated patterns discovery, and dense image alignment. All the tasks are very challenging from a computer vision point of view, main difficulties include: i) no annotations are available so that the approach we develop needs to be *unsupervised*; ii) there are differences in the artistic media (oil, pastel, drawing, etc) of the different depictions we want to analyze, so we face *cross-domain* problems.

Our first task, *style-invariant patterns retrieval* aims at finding relevant motifs in a collection of artworks similar to a given query motif. The retrieved motifs should depict the same content but might have different styles. The task is motivated by several image search applications in art images and historical documents. Beyond heritage images, image retrieval has been widely used in image search tools such as Google Image Search [1].

Our second task, *near-duplicated patterns discovery* aims at identifying repeated visual patterns in a collection of artworks. The task is motivated by finding visual correspondences among artworks, which might indicate authorship and provenance. Beyond art history, near-duplicated pattern discovery can be used to discover common objects in generic image collection, which can be useful to ease data annotation processes.

Our third task, *dense image alignment* aims at predicting pixel-level alignment between images. Our goal is to compare two paintings or details of two artworks depicting the same content but with different styles. On natural images, optical flow is also beneficial to two-view geometry estimation and 3D reconstruction.

This thesis includes three technical contributions.

Our first contribution is a self-supervised approach to adapt standard deep features for cross-domain matching by fine-tuning it on the specific art collection. More precisely, we use spatial consistency between neighboring feature matches as supervision. The adapted features lead to more accurate style-invariant matching. We leverage multi-resolution feature matching and geometric verification to identify near-duplicated patterns in the dataset. Along with the approach, we also propose a

[1] https://www.google.com/imghp?hl=fr

dataset which we refer to as the Brueghel (Shen et al., 2019; bru) dataset which allows evaluating one-shot cross-domain art detail detection.

Our second contribution is learning co-segmentation for a pair of images on a synthetic dataset. We generate training pairs by blending objects into a background image and applying style transfer to the resulting composition. As a result, we have access to ground-truth masks of the blended objects and correspondences between the original and synthesized image. We empirically compare the performances obtained with two networks: Sparse Nc-Net (Rocco et al., 2020) and a transformer-based (Vaswani et al., 2017) architecture, and show that employing Poisson Blending (Pérez et al., 2003) and style transfer is crucial for generalization. In terms of results, the trained transformer improves performance on near-duplicated art detail detection and discovery. Additionally, we show that our approach performs well on place recognition and object discovery in natural images.

Our last contribution is a two-stage method for generic image alignment. In the coarse stage, we estimate Homography transformations between a pair of images using standard deep feature matches and RANSAC; in the fine stage, we design and learn a Convolutional Neural Network (CNN) to predict pixel-level alignment using a reconstruction loss. The proposed approach is simple and allows aligning near-duplicated art patterns. On natural images, the proposed approach shows competitive or better performance across different tasks: optical flow estimation, sparse correspondences evaluation, two-view geometry estimation, 3D reconstruction, and aligning Internet images.

Résumé

L'objectif de cette thèse est de développer des méthodes d'apprentissage profond pour l'analyse d'œuvres d'art. Plus précisément, nous nous concentrons sur trois tâches particulières : la recherche de motifs invariants de style, la découverte de motifs quasi-dupliqués et l'alignement dense d'images. Toutes les tâches sont très difficiles du point de vue de la vision par ordinateur, les principales difficultés incluent : i) aucune annotation n'est disponible, de sorte que l'approche que nous développons doit être *non-supervisée*; ii) il existe des différences dans les supports artistiques (huile, pastel, dessin, etc.) des différentes représentations que nous voulons analyser, nous sommes donc confrontés à des problèmes *inter-domaines*.

Notre première tâche, *style-invariant patterns retrieval* vise à trouver des motifs pertinents dans une collection d'œuvres d'art similaires à un motif de requête donné. Les motifs récupérés doivent représenter le même contenu mais peuvent avoir des styles différents. La tâche est motivée par plusieurs applications de recherche d'images dans des images d'art et des documents historiques. Au-delà des images patrimoniales, la recherche d'images est largement utilisée par exemple dans les outils de recherche d'images tels que Google Image Search [2].

Notre deuxième tâche, *découverte de motifs quasi-dupliqués* vise à identifier des motifs visuels répétés dans une collection d'œuvres d'art. La tâche est motivée par la recherche de correspondances visuelles entre les œuvres d'art, qui pourraient indiquer la paternité et la provenance. Au-delà de l'histoire de l'art, la découverte de motifs similaires peut être utilisée pour découvrir des objets communs dans une collection d'images naturelles, ce qui peut être utile, par exemple, pour faciliter le processus d'annotation des données.

Notre troisième tâche, *alignement dense d'images* vise à prédire l'alignement au niveau des pixels entre les images. Notre objectif est de comparer deux tableaux ou des détails d'œuvres d'art représentant le même contenu mais avec des styles différents. Sur les images naturelles, le flux optique est également bénéfique pour l'estimation de la géométrie à deux vues et la reconstruction 3D.

Cette thèse comprend trois contributions techniques.

Notre première contribution est une approche auto-supervisée pour adapter un

[2] https://www.google.com/imghp?hl=fr

descriptor standard de reseau de neurones pour la correspondance entre domaines en fine-tuning sur la collection d'art spécifique. Plus précisément, nous utilisons la cohérence spatiale entre les correspondances de caractéristiques voisines comme supervision. Le descriptor de reseau de neurones adapté conduit à une correspondance invariante de style plus précise. Nous utilisons de la correspondance de multi-résolution et de la vérification géométrique pour identifier des modèles presque dupliqués dans l'ensemble de données. Parallèlement à l'approche, nous proposons également un ensemble de données que nous appelons Brueghel (Shen et al., 2019; bru) qui permet d'évaluer la détection des détails de l'art inter-domaines.

Notre deuxième contribution est l'apprentissage de la co-segmentation pour une paire d'images sur un jeu de données synthétique. Nous générons des paires d'entraînement en mélangeant des objets dans une image d'arrière-plan et en appliquant un transfert de style à la composition résultante. En conséquence, nous avons accès aux masques de vérité terrain des objets mélangés et aux correspondances entre l'image originale et l'image synthétisée. Nous comparons empiriquement les performances obtenues avec deux réseaux : Sparse Nc-Net (Rocco et al., 2020) et une architecture à base de transformer (Vaswani et al., 2017), et montrons que l'utilisation de Poisson Blending (Pérez et al., 2003) et de transfert de style est cruciale pour la généralisation. En termes de résultats, le transformer entraîné améliore les performances de détection et de découverte des détails d'art presque dupliqués. De plus, nous montrons que notre approche fonctionne bien sur la reconnaissance de lieux et la découverte d'objets dans les images naturelles.

Notre dernière contribution est une méthode en deux étapes pour l'alignement d'images génériques. Au stade grossier, nous estimons les transformations d'homographie entre une paire d'images en utilisant des correspondances de descriptors standard de reseau de neurones et RANSAC ; au stade avancé, nous concevons et apprenons un réseau de neurones convolutifs (CNN) pour prédire l'alignement au niveau des pixels à l'aide d'une perte de reconstruction. L'approche proposée est simple et permet d'aligner des motifs artistiques presque dupliqués. Sur les images naturelles, l'approche proposée montre des performances compétitives ou meilleures sur différentes tâches : estimation de flux optique, évaluation de correspondances creuses, estimation de géométrie à deux vues, reconstruction 3D et alignement d'images Internet.

Acknowledgments

This thesis would not have been possible without the help and nurturing of many people.

I would like to express my deepest gratitude to my advisor, Mathieu, for your patience, guidance, and support. I have greatly benefited from your wealth of knowledge and meticulous editing. I am very grateful that you took me on as a student and continued to have faith in me over the years. I am also extremely thankful to Alyosha and Armand. I learn a lot from you and feel extremely lucky to work with you. I enjoyed all the discussions with you.

I would like to warmly thank all my committee members. It was not easy to organize the defense across different time zones. Thanks for everyone's effort to make my defense possible. Special thank Prof.Gousseau and Prof.Heutte for the detailed comments on the manuscript. I am also grateful to Dr.Ginosar and Dr.Pérez for having accepted to be part of the jury.

I am fortunate to have been a part of the Imagine group, where I greatly enjoyed my life as a Ph.D. student. Thanks for everyone's effort to create a fantastic lab atmosphere. Some moments have been deeply imprinted on my memory: hiking, football, beer, conference, Christmas meal, etc. Thanks to David and Vincent for leading the lab and handling all the issues of the lab. Thanks to Guillaume, Renaud, Gül, Pascal, Bertrand and Chaohui for all the feedbacks, reviews of my works as well as joyful discussions during coffee sessions. Thanks to Thibault (for the crazy California camping and a lot of things), Theo (for hosting me as a family member), François (for the great collaboration), Abdou (for your patience to answer all my stupid questions), Tom (for organizing foot, student meetings and everything), Georgy (for being the nicest office mate), Yanis (for all the ideas we had brainstormed), Othman and Pierre-Alain (for always being with me in ENPC), Yang (for always waiting for me until 10pm to leave the office), Yumin (for tasty chinese dishes), Thomas (for your kindness

to help everyone including me in the lab), Xuchong (for the excellent trip in Ile de Ré), Shell (for treating me as a younger brother and helping me on developing research ideas), Simon (for all the cool stuffs you taught me and your excellent football skills), Nguyen (for sharing fascinating vietnamese culture), Michaël (for all the interesting experiences shared with me and valuable comments on my works), Robin (for sharing your only machine with me and being extremely nice to me even I made stupid bugs on the machine), Oumayma (for the great project working together and your interests on chinese culture), Nicolas (for lots of useful suggestions and deep communications), Mathis (for your tips on my works and efforts you made to share knowledge on reading groups), Elliot (for being my excellent foot teammate), Yue (for helping me searching the apartment), Rahima (for being extremely nice to me and sharing all the complicated administrative information with me), Victor (for funny jokes during beer sessions), Romain (for sharing your practical working tools and experiences), Hugo (for tips on interns and interviews), Giorgia (for working, playing and doing sports together). I am not a very talkative person and would like to take this opportunity to thank everyone. I had a lot of great time in lovely Paris.

I also would like to thank my friends Yuan, Yuxiang, Xijun, Tong, Ling, Yi, Yuelu, Dexiong, Huiqin, and many others. Thanks for supporting me, despite the distance.

Last but not least, I want to express special thanks to my grandfather and my parents. Thanks for your love and the courage you had to bring me to this world. Thanks to my brother and my girlfriend. Thank you for your unfailing support and understanding.

Contents

Chapter 1

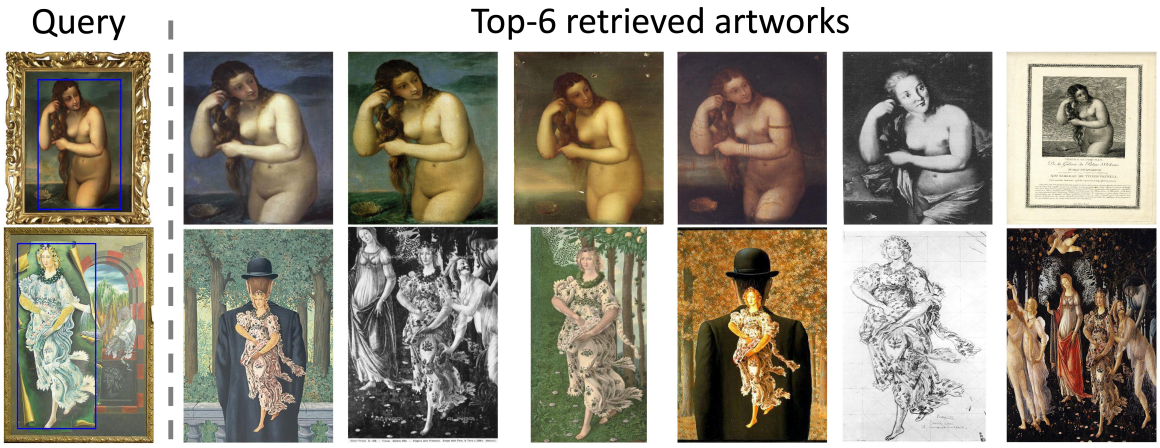
Introduction

1.1 Goals

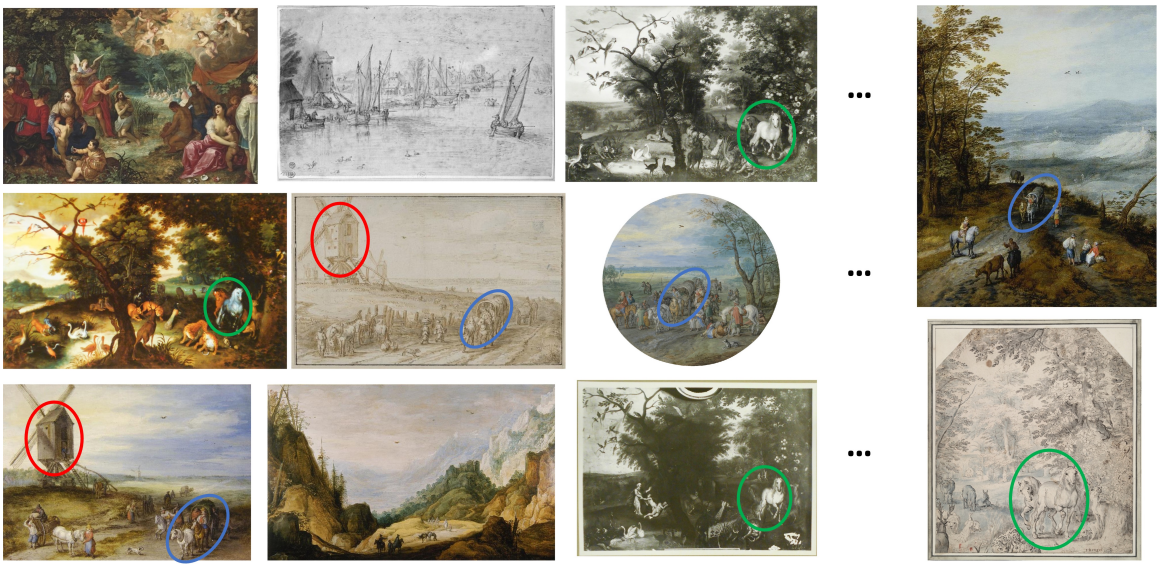
When creating an artwork, it is common for an artist to reuse the same visual elements or motifs. However, the different versions might be differ in terms of color, style, artistic media, and geometric deformation. Identifying repeated patterns yield insights on artworks circulations and provenance, which is important for Art Historians. The goal of this thesis is to develop deep learning methods for artwork analysis with a focus on three tasks: (i) style-invariant patterns retrieval, (ii) discovering near-duplicated visual patterns in a collection of artworks, (iii) dense image alignment. These tasks are illustrated in Fig. 1.1.

Style-invariant patterns retrieval aims at retrieving motifs similar to a given query pattern. The relevant motifs have the same content, whereas styles and context might be different. An example from the Venus collection (Shen et al., 2022) is shown in Fig. 1.1a. We can see the top retrieved artworks contain the same motif as the queries but have a very different appearance.

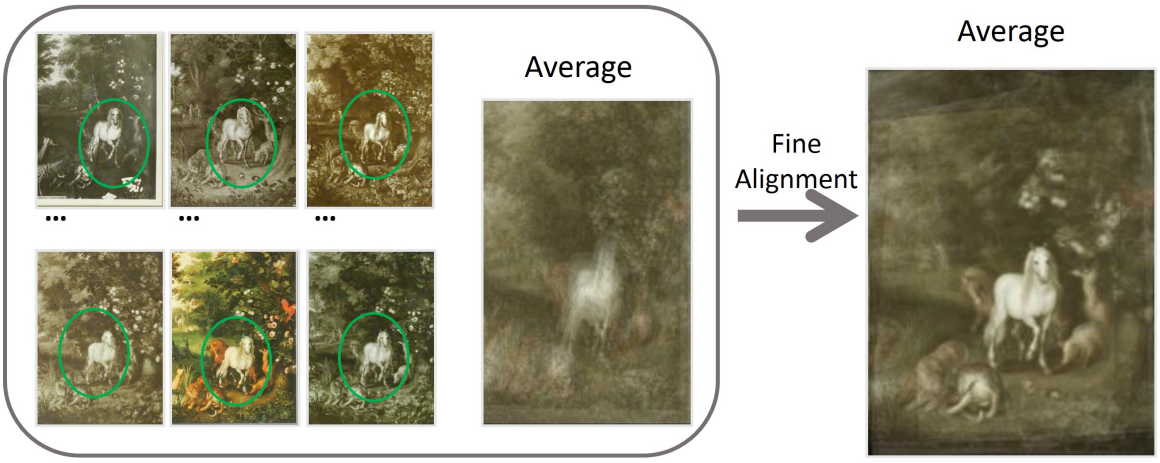
Near-duplicated patterns discovery aims at identifying near-duplicated patterns in a collection of images independently of any queries. Unlike image clustering, we do not assume that every image in the collection is associated to a unique pattern and we also aim at localizing the repeated patterns. An example on the Brueghel (Shen et al., 2019; bru) dataset is given in Fig. 1.1b. We are interested in identifying the groups of repeated visual details (shown in red, blue and green circle),



(a) **Style-invariant patterns retrieval**. Our first goal is to retrieve artworks depicting the same content as given query motifs (blue bounding boxes).



(b) **Near-duplicated patterns discovery**. Our second goal is to identify near-duplicated visual patterns (red, blue and green circle) in a collection of artworks.



(c) **Dense image alignment**. Our third goal is to estimate pixel-level alignment for patterns that are near-duplicated and are depicted with different styles.

Figure 1.1: The three main tasks addressed in this thesis

they might be present in only a couple of images, an image may contain different details and most images do not contain any.

Dense image alignment aims at estimating pixel-level alignment between images that depict similar content. We design a generic image alignment algorithm, with a specific focus on the capacity to align images from different visual domains, e.g. artwork depicting the same content with different styles. An example on the Brueghel (Shen et al., 2019; bru) dataset can be seen in Fig. 1.1c.

The above three tasks can be complementary for artwork analysis. On artwork analysis, if queries are available, we could retrieve artworks containing relevant motifs and focus on analyzing on top retrieved artworks. If no queries are available, we could perform discovery on the dataset and study the discovered clusters. For both cases, detailed analysis is possible through dense alignment.

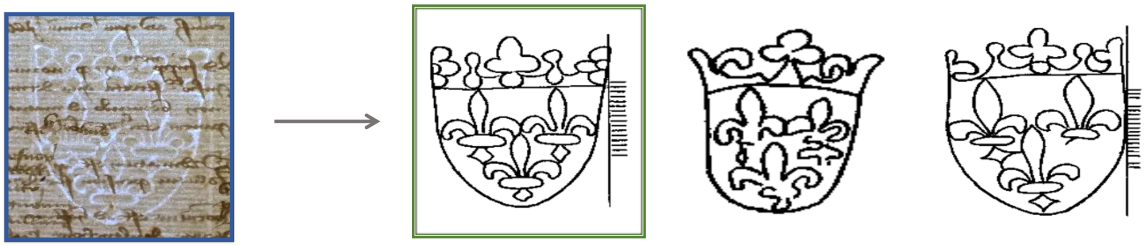
1.2 Motivations

Although this thesis mainly focuses on artworks analysis, the three tasks addressed in the thesis are core computer vision problems. They are also motivated by a number of applications beyond artworks analysis.

Image retrieval. Retrieving relevant images is an important problems in computer vision. The task has several applications I tackled during my PhD but are not discussed in details in this thesis: place recognition (Shen et al., 2021a), historical watermark recognition (Shen et al., 2020b), sketch-based image retrieval (Shen et al., 2020b) and matching illustrations in copied manuscripts (Kaoua et al., 2021):

- **Place recognition** (Fig. 1.2a, (Shen et al., 2021a)). Given a query image and a database of geolocated images, the goal is to retrieve an image of the same place. One challenge is that the images can have very different appearance, e.g. images taken during day-time and night-time.
- **Historical watermark recognition** (Fig. 1.2b, (Shen et al., 2020b)). Queries are photographs of historical documents and the searching database consists of drawings of watermarks, the goal is to identify the watermark in the database

(a) **Place recognition.** Given a query image and a database of geolocated images, the goal is to retrieve an image of the same place.



(b) **Historical watermark recognition.** Given a query historical document and a database of drawings of watermarks, the goal is to identify the drawing in the database which corresponds to the historical document.

(c) **Sketch-based image retrieval.** Given a query sketch and a database of natural images, the goal is to retrieve natural images correspond to the content of sketch.

(d) **Matching illustrations in copied manuscripts.** Given illustrations from copied manuscripts, the goal is to match illustrations in manuscripts.

Figure 1.2: Applications of **image retrieval**.

which can be seen in the photograph. The watermark provides clues to date and locate paper, which is useful for historians. The main challenges include fine-grained search, lack of annotations and the fact that watermarks in the query photographs have a very different appearance compared to the drawings in the database.

- **Sketch-based image retrieval** (Fig. 1.2c, (Shen et al., 2020b)). Given a query sketch and a database of natural images, the goal is to retrieve natural images correspond to the content of sketch. The application can be a tool for image search when looking for a specific instance of an object, e.g. a specific type of shoes, because it is easier to draw characteristics than describe them. The task is challenging as the search is fine-grained, cross-domain and some details might not be drawn in the query sketches.

- **Matching illustrations in copied manuscripts** (Fig. 1.2d, (Kaoua et al., 2021)). Historical manuscripts often exist in different versions made by different copyists and illustrations might even be re-used in different texts. The goal of this task is to find corresponding illustrations in several manuscripts. In the case of scientific manuscripts (e.g. botanical), retrieving corresponding illustrations might help historians to understand the evolution of scientific knowledge. The task is difficult because the order of the illustrations might differ in different manuscripts and content and styles might be changed in corresponding illustrations.

Object discovery. The task is to automatically discover repeated objects in a dataset without any human annotations. Most visual analysis deep learning works rely on annotations (He et al., 2016; Zhou et al., 2018), which are costly. It would be much more efficient to be able to directly analyze a collection of images and identify the repeated objects. The main application of object discovery is to automatically collect dataset annotations.

Imagine for example we want to collect segmentation annotations of cars. First, we can search relevant images correspond to a key word “Car" through an image search engine such as Google Search [1]. The gathered images are noisy i.e. it might

[1] https://www.google.com.fr/imghp?hl=fr&ogbl

Figure 1.3: Automatic dataset annotation is one of the motivation for **object discovery**. Object discovery allows automatically collecting annotations from Internet images. Examples are from (Rubinstein et al., 2013). A noisy dataset can be gathered via Internet searching engine then discovering common objects can remove distractors and obtain segmentation annotations.

include irrelevant images such as images of wheel, logos of cars etc. An image discovery algorithm should be able to take this noisy non-annotated dataset, identify the relevant car images and even provide pixel-level annotations. A classical approach to this task is the one of (Rubinstein et al., 2013), illustrated on figure 1.3.

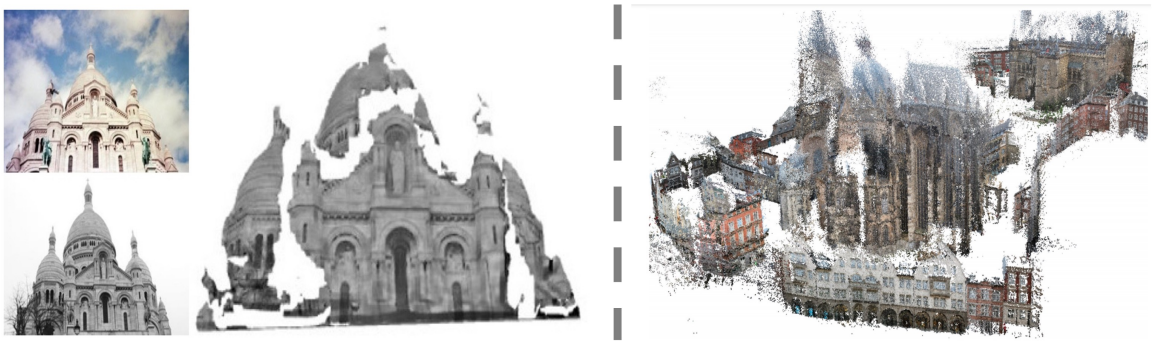
Dense image alignment. Dense alignment of images, also known as the optical flow problem, is a classical problem in computer vision. Precise optical flow leads to numerous applications, shown in Fig. 1.4:

- **Autonomous driving** (Fig. 1.4a, (Geiger et al., 2012)). Estimating the optical flow between images taken by cameras in a vehicle could be leveraged for autonomous driving applications. The input images often have little appearance difference and the dense optical flow yield insights into the geometric layout of a scene as well as its decomposition into individually moving objects.

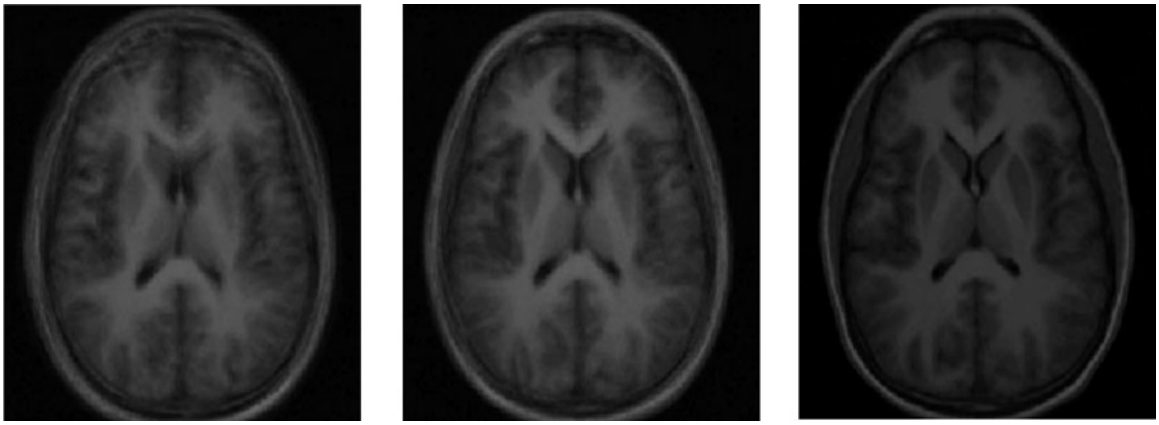
- **3D reconstruction** (Fig. 1.4b, (Shen et al., 2020a; Truong et al., 2021a)). Dense optical flow can be useful for two-view geometry estimation and reconstructing meaningful 3D models from a pair of images (Shen et al., 2020a) (Fig. 1.4, left). Dense optical flow can also replace keypoints matches and be used in multi-view 3D reconstruction (Truong et al., 2021a) (Fig. 1.4, right).

- **Medical image alignment** (Fig. 1.4c, (Hill et al., 2001)). In medical applications, multiple medical images can be acquired at different times. To compare

(a) **Autonomous driving.** Optical flow (2nd row) provides insight to segment moving objects (1st row). Images from (Menze and Geiger, 2015b).



(b) **3D reconstruction.** Dense flow field between a pair of images allows 3D reconstruction (left, two-view geometry estimation (Shen et al., 2020a)). Dense flow can also replace keypoint matches for multi-view 3D reconstruction (right, 3D reconstruction of Aachen (Sattler et al., 2018; Truong et al., 2021a)) .



(c) **Medical images alignment.** Axial slices from average images produced from MR scans, alignment using rigid registration (left), affine registration (middle) and non-rigid registration (right). Figures from (Hill et al., 2001).

Figure 1.4: Example of applications of **dense image alignment**.

the images and analyze their evolution, it would be useful to be able to align densely the images.

1.3 Challenges

To solve our tasks, we will face three important challenges: lack of supervision, handling domain gap between the depictions, and a scalability issue. These challenges are illustrated in Fig. 1.5.

Lack of supervision. One critical challenge is that we cannot assume any annotations are available. Indeed, it is expensive and time consuming to collect annotations. For patterns retrieval, the annotations need to be specific to a collection and it is hard for a human expert to find the same motifs when the size of the collection becomes large. For discovery, the annotation process is even more difficult, as one needs to identify all repeated patterns in a collection and the annotations should include segmentation. For dense image alignment, the annotations would need to be done pixel-wise. Indeed, the defomations happenning during the copy process are far from being parametric or rigid, for example in the example of Fig. 1.5a, the three characters groups are clearly moving with respect to each other. Note that collecting annotations is also a general challenge for many other applications in vision, such as object detection (Lin et al., 2014), semantic segmentation (Zhou et al., 2017) and 3D reconstruction (Li and Snavely, 2018) etc.

Diversity of depiction styles. For artists, it is common to reuse the same visual patterns. However, the depiction styles might be different. An example is shown in Fig. 1.5b, the same content (*Man on Egg*) is repeated in four artworks but depicted using different media and with different styles. Note that handling domain gap is also an important and general challenge in computer vision. Related active research topics include domain adaptation (Saenko et al., 2010) and domain generalization (Li et al., 2017).

Scalability. We also face a scalability issue for patterns retrieval and discovery problem. For retrieval, we aims at retrieving the same patterns at any scales in a large dataset. For discovery, in addition to the scale issue, the repeated patterns might be present only in a couples of images in a large collection of artworks. An

(a) **Difficulty of annotations**. Alignment between related artworks cannot be well approximated by a simple parametric transformation. For example, in this two versions of *Diana and Actaeon* in the Brueghel (Shen et al., 2019; bru) dataset the displacements are very complex, and their alignment would require an extremely tedious pixel-level annotations of the correspondences.

(b) **Diversity of depiction styles.** Four paintings on the topic *Man on Egg* in the Brueghel (Shen et al., 2019; bru) dataset.

(c) **Scalability**. We want to identify repeated patterns (horse in green circle) in a large dataset that could have any scales in the images, which is very computationnaly difficult.

Figure 1.5: **The three challenges of our tasks**: lack of supervision (Fig. 1.5a), diversity of depiction styles (Fig. 1.5b) and scalability issue. (Fig. 1.5c).

example is shown in Fig. 1.5c, the horses (in green circle) are repeated but with very different scales in the images, and they are present only in a few images of a large dataset ($\sim$ 20 occurences over 1,587 images). Our algorithm thus has to scale to large datasets, without missing rare occurence of small details.

1.4 Contributions

In this thesis, we have three main technical contributions.

Self-supervised style-invariant feature learning. Our first contribution is a self-supervised feature fine-tuning strategy to address style-invariant feature matching. Our key insight is leveraging spatial consistency to identify positive and negative training pairs. The training pairs are then used to optimize a standard triplet loss, which leads to features adapted to the training set for correspondences. We show that the proposed approach enables matching features across different domains. We validate our fine-tuning strategy on artwork datasets as well as geo-localization datasets.

Learning co-segmentation from synthetic data. Our second contribution is an approach of learning co-segmentation from synthetic data. To generate the synthetic data, we blend objects to background images using style transfer and Poisson blending Pérez et al. (2003). We also propose a transformer-based architecture (Vaswani et al., 2017) for co-segmentation. We demonstrate that it is important to include correspondences in the objective function. The approach is validated on various tasks including one-shot art detail detection and place recognition.

Unsupervised two-stage robust dense image alignment. Our last contribution is an unsupervised approach for robust dense image alignment. Our approach contains two stages: in the first stage, we estimate Homography transformations based on standard multi-scale feature matching and RANSAC (Fischler and Bolles, 1981); in the second stage, we learn a Convolutional Neural Network (CNN) through the optimization of a reconstruction loss between target images and warped source images. The entire pipeline is unsupervised. We show that our approach performs well on various tasks, including standard optical flow estimation, sparse correspondences evaluation, two-view geometry estimation, and 3D reconstruction.

1.5 Thesis Outline

This thesis is organized as follows.

Chapter 2: Related Work. We first review deep learning-based approaches for artworks analysis in Sec. 2.1. In Sec. 2.2, we study related works on image retrieval. We then provided analysis on object discovery and co-segmentation approaches that are related to this thesis in Sec. 2.3. Finally, we review deep-learning based approach for dense image alignment in Sec. 2.4.

Chapter 3: Learning Features for Artworks Analysis. This chapter presents the first contribution of this thesis : our self-supervised feature fine-tuning strategy. We explain the feature learning strategy in Sec. 3.3. We leverage it to perform object discovery using a simple approach based on multi-scale feature matching and RANSAC (Fischler and Bolles, 1981), which is detailed in Sec. 3.4. In Sec. 3.5, we introduce the Brueghel dataset (Shen et al., 2019) and present experimental results of the feature fine-tuning strategy and of the discovery approach on various art image datasets as well as geo-localisation datasets.

Chapter 4: Learning Co-segmentation for Art Pattern Discovery. This chapter is about the second contribution of this thesis: learning co-segmentation from synthetic data. Our training data generation, as well as two architectures, are detailed in Sec. 4.3. In Sec. 4.4, we present how to employ the output of the networks to perform image retrieval. The approach is validated on two completely different tasks: one-shot art detail detection and place recognition. We finally show how to leverage the output of the networks for object discovery with using classical spectral clustering on a correspondence graph in Sec. 4.5.

Chapter 5: Unsupervised Dense Image Alignment. This chapter includes the last contribution of this thesis: Unsupervised two-stage robust dense image alignment. In Sec. 5.3, we present our two-stage image alignment approach. Beyond qualitative results on artworks alignment, we validate the approach on various tasks including: optical flow evaluation (Sec. 5.4.1), sparse correspondences evaluation (Sec. 5.4.1), two-view geometry estimation (Sec. 5.4.2) and day-night visual Localization (Sec. **??**). With the precise optical flow, we also explore several applications in Sec. 5.4.3, such

as: texture transfer and Internet images alignment.

Chapter 6: Conclusion. We conclude with a summary of contributions of this thesis, discussion and suggestions on future research directions.

1.6 Publication List

This thesis presents the contribution made in four papers (Shen et al., 2019, 2021a, 2020a, 2022):

- **Xi Shen**, Alexei A. Efros, Mathieu Aubry.
  Discovering Visual Patterns in Art Collections with Spatially-consistent Feature Learning.
  Code & Project page : http://imagine.enpc.fr/~shenx/ArtMiner/
  In *Computer Vision and Pattern Recognition (CVPR)*, 2019.

- **Xi Shen**, Alexei A. Efros, Armand Joulin, Mathieu Aubry.
  Learning Co-segmentation by Segment Swapping for Retrieval and Discovery.
  Code & Project page : http://imagine.enpc.fr/~shenx/Coseg/
  In *submission*, 2021.

- **Xi Shen**, François Darmon, Alexei A. Efros, Mathieu Aubry.
  RANSAC-Flow: Generic Two-stage Image Alignment.
  Code & Project page : http://imagine.enpc.fr/~shenx/RANSAC-Flow/
  In *European Conference on Computer Vision (ECCV), 2020*

- **Xi Shen**, Robin Champenois, Shiry Ginosar, Ilaria Pastrolin, Morgane Rousselot, Oumayma Bounou, Tom Monnier, Spyros Gidaris, François Bougard, Pierre-Guillaume Raverdy, Marie-Françoise Limon, Christine Bénévent, Marc Smith, Olivier Poncet, K. Bender, Joyeux-Prunel Béatrice, Elizabeth Honig, Alexei A. Efros, Mathieu Aubry
  Spatially-consistent Feature Matching and Learning for Art Collections and Watermark Recognition.
  Code & Project page : http://imagine.enpc.fr/~shenx/HisImgAnalysis/
  In submission, minor revision in *International Journal of Computer Vision (IJCV)*, 2022.

We open-sourced and maintained implementations corresponding to the four papers [2], the implementation of ArtMiner (Shen et al., 2019) and RANSAC-Flow (Shen et al., 2020a) on GitHub received 100 and 307 stars respectively.

We also extend the discovery score in the first publication (Shen et al., 2019) to weakly supervised watermark recognition and unsupervised image collation, which led to the following publications (Shen et al., 2020b; Bounou et al., 2020; Kaoua et al., 2021):

- **Xi Shen**, Ilaria Pastrolin, Oumayma Bounou, Spyros Gidaris, Marc Smith, Olivier Poncet, Mathieu Aubry.
  Large-Scale Historical Watermark Recognition: dataset and a new consistency-based approach.
  Code & Project page : http://imagine.enpc.fr/~shenx/Watermark/
  In *International Conference on Pattern Recognition (ICPR)*, 2020.

- Oumayma Bounou, Tom Monnier, Ilaria Pastrolin, **Xi Shen**, Christine Bénévent, Marie-Françoise Limon-Bonnet, François Bougard, Mathieu Aubry, Marc Smith, Olivier Poncet, Pierre-Guillaume Raverdy.
  A Web Application for Watermark Recognition.
  Web app : https://filigranes.inria.fr/#/filigrane-search
  In *Journal of Data Mining and Digital Humanities*, 2021.

- Ryad Kaoua, **Xi Shen**, Alexandra Durr, Stavros Lazaris, David Picard, Mathieu Aubry
  Image Collation: Matching illustrations in manuscripts.
  Code & Project page : http://imagine.enpc.fr/~shenx/ImageCollation/
  In *International Conference on Document Analysis and Recognition (ICDAR), 2021*

During my Ph.D., I also worked on learning with weak supervision (weakly supervised learning) and few samples (few-shot learning), which results in the following publications which are not directly related to the work presented in this thesis (Yuan et al., 2019; Hu et al., 2020; Shen et al., 2021b):

- **Xi Shen**, Yang Xiao, Shell Xu Hu, Othman Sbai and Mathieu Aubry.
  Re-ranking for Image Retrieval and Transductive Few-shot Classification.

[2] https://github.com/XiSHEN0220

Code & Project page : http://imagine.enpc.fr/~shenx/SSR/
In *Conference on Neural Information Processing Systems (NeurIPS)*, 2021.

- Yuan Yuan, Lyu Yueming, **Xi Shen**, Tsang Ivor W and Yeung Dit-Yan.
  Marginalized Average Attentional Network for Weakly-supervised Learning.
  Code : https://github.com/yyuanad/MAAN
  In *International Conference on Learning Representations (ICLR)*, 2019.

- Shell Xu Hu, Pablo Moreno, Yang Xiao, **Xi Shen**, Guillaume Obozinski, Neil Lawrence, and Andreas Damianou.
  Empirical Bayes Transductive Meta-learning with Synthetic Gradients.
  Code : https://github.com/hushell/sib_meta_learn
  In *International Conference on Learning Representations (ICLR)*, 2020.

Chapter 2

Related Work

2.1 Deep Learning for Artwork Analysis

There is a long-standing and fruitful collaboration between computer vision and art. For artists and art historians, deep learning becomes a practical tool to analyze and create art. In this section, we discuss deep learning approaches on four topics which are related to artworks analysis: i) artwork retrieval and artistic influence discovery; ii) artwork attributes classification; iii) object detection in artworks; iv) creating art. These topics outline some main research trends and directions.

Artwork retrieval and artistic influence discovery. An important task for art historians is identifying similarity relationships between artworks of different artists and painting schools. These relationships enable art historians to discover the provenance and influences from an artistic movement to another (David, 2019). Previous works on this topics can be roughly divied into two directions: i) artwork retrieval: when a query artwork is given and it is required to retrieve relevant artworks; ii) artwork clustering: we are asked to analyse relationships of artworks in a dataset without using any annotations.

For artwork retrieval, based on traditional handcraft features, (Shrivastava et al., 2011) computed visual similarity between images cross different domains and show that the proposed algorithm worked also for art images. In deep era, (Crowley et al., 2015) aimed at matching photos of a person to paintings of the same person. It showed that Convolutional Neural Network (CNN) representations perform clearly better than traditional descriptors. Training on manually annotated artworks, (Seguin

et al., 2016) has shown that a pre-trained CNN performs better than a classic bag-of-words method in predicting whether pairs of paintings are visually related to each other. DeepArt (Mao et al., 2017) learned visual art representations for retrieval by a triplet-based deep ranking method. (Dovgalecs et al., 2013) proposed a system for patterns spotting in historical documents, which aimed at detecting the same patterns, based on a user made query. The system employed a two-stage approach: an offline feature extraction based on SIFT (Lowe, 2004) and bag-of-words features and an online query search phase. The performance can be improved through a siamese neural network (Wiggers et al., 2019) and feature pyramid networks (Ubeda et al., 2020). Beyond the advancement in research, several projects have built search engines for artworks (img; tim).

For artwork clustering, (Gultepe et al., 2018) employed k-means on the top of features which were then fed into a spectral clustering algorithm to group paintings. Similarly, (Castellano and Vessio, 2020; Castellano et al., 2021; Castellano and Vessio, 2021) proposed a method aimed at finding visual links among paintings in a completely unsupervised way. Their methods are built upon visual attributes learned by a deep pre-trained model. Furthermore, they showed that influences between painters could be exploited from the retrieved paintings.

Artwork attributes classification. In visual art, another research direction aims at recognizing some artwork attributes such as artist, genre, period, etc. A pioneer work is (Karayev et al., 2014), which showed that a pre-trained CNN is able to predict the painting school of an artwork. Beyond the quantitative results, (Karayev et al., 2014) also observed that the predicted style of a painting is highly correlated with the content of the painting. (Van Noord et al., 2015) focused on the artist attribution task, which aimed at attributing an unseen artwork to the artist who created it. (Van Noord et al., 2015) trained on a large collection of paintings using AlexNet (Krizhevsky et al., 2012). Interestingly, for paintings with multiple authorship, it is possible to distinguish areas created by different artists via visualizing the importance of regions for the predictions. (Saleh and Elgammal, 2015) investigated different features (classical and CNN features) and learned to predict a painting's style, genre, and artist. (Tan et al., 2016) aimed at solving the same tasks, and considered training on each task individually. They observed that all the tasks can benefit from CNN models pre-trained on ImageNet (Deng et al., 2009). (Saleh et al., 2016) trained a style classification model and obtained the most appropriate features to classifier styles by

empirically validated on artwork datasets for the style classification. (Chen and Yang, 2019) showed that combining features from both lower and higher layers improved the performance of style classification. OmniArt (Strezoski and Worring, 2017) proposed to solve multiple attributes predictions directly through a multi-output CNN model. The overall training is carried out by minimizing an aggregated loss obtained as a weighted combination of the separate losses.

Object detection in artworks. Several works aim at finding out when a specific object first appeared in a painting or how the representation of an object evolved over time. A pioneering work is (Crowley and Zisserman, 2014), which proposed a system that retrieves training samples given an input query by crawling Google Images on-the-fly. These are then used to learn a classifier. The final output of the classifier is a ranked list of paintings containing the queried object. The approach has been improved by moving from image-level classifiers to object detection systems (Crowley and Zisserman, 2016). (Yin et al., 2016) used the same Brueghel data (bru; Honig, 2016) as us and annotate it to train detectors for five object categories (carts, cows, windmills, rowboats, and sailboats). (Westlake et al., 2016) focused on detecting people in artworks, they showed that deep features trained on natural images lead to severe overfitting issue and fine-tuning on artwork dataset achieves clearly better generalization. Similarly, (Ginosar et al., 2014) aimed at detecting people but with a focus on Cubist paintings and empirically found that the deformable partbased models (DPM) outperforms other classical detectors such as RCNN (Girshick et al., 2014). (Gonthier et al., 2018) proposed a weakly supervised approach that can learn to detect objects based only on image-level annotations. (Crowley and Zisserman, 2013) started from a large dataset of images of vases with text descriptions and aimed at detecting gods and animals. It developed a weakly supervised learning approach to solve the correspondence problem between the descriptions and unknown image regions.

Different from detecting objects of pre-defined categories, one-shot detection aims at detecting visual details that are similar to given queries details.The task is interesting for art historians, as it is possible to track the circulation of these visual patterns over long spatial and temporal migrations, as they are progressively copied by several generations of painters. (Seguin et al., 2017) detected duplicated art details with using CNN features and sliding window proposals. Beyond matching 2D art details, (Aubry et al., 2014) showed that it is possible to align non-photographic

depictions to their 3D model. More precisely, (Aubry et al., 2014) proposed to learn a set of visual elements that match in 2D depictions of the scene despite large variations in rendering style (e.g. watercolor, sketch, historical photograph) and structural changes (e.g. missing scene parts, large occluders) of the scene. The proposed visual elements allowed automatically finding an approximate viewpoint of paintings with respect to a 3D model of the site.

Creating art. On the synthesis side, promising results have been obtained for transferring artistic style to a photograph. (Gatys et al., 2015) found that optimizing Gram matrix can be used to generate texture. Based on this idea, (Gatys et al., 2016) first explored transfer the style of one image (style image) to another image(content image). The output stylized image is obtained through optimizing two distances: features between the generated image and the content image and the Gram matrix between the generated image and the style image. However, as the stylized image is obtained by iterative optimization until it matches the desired statistics, t (Gatys et al., 2016) is computationally inefficient. To address the issue, (Johnson et al., 2016) proposed to learn an image transformation network to transform input images into output images through a perceptual loss, which is similar to (Gatys et al., 2016). Beyond the task of style transfer, the proposed method is shown to be effective on single-image super-resolution. Instead of optimizing the Gram matrix, the image translation problem can also be properly addressed using GAN (Goodfellow et al., 2014). Cycle-GAN (Zhu et al., 2017) is a representative work, which leverages cycle-consistency between two domains without image-pair annotations. (Elgammal et al., 2017) even tried to create art. (Elgammal et al., 2017) proposed Creative Adversarial Network (CAN): a variant of a classic GAN which generated novel artworks by maximizing the deviation from established styles and minimizing the deviation from art distribution. The generated art was assessed by human evaluators, who regularly confused the generated art with human art.

2.2 Image Retrieval

Given a query image, image retrieval searches for similar images in a set of test images. In this section, we review some approaches from using handcrafted features to deep learning techniques.

Handcrafted features. Video Google (Sivic and Zisserman, 2003) consists of three steps for image retrieval: i) computing SIFT keypoints (Lowe, 2004); ii) building a visual vocabulary by K-means clustering with k clusters (visual words); iii) the collection of visual words are used in Term Frequency Inverse Document Frequency (TF-IDF) scoring of the relevance of an image to the query. More precisely, the image d is represented by a k-vector $V_d = [t_1, t_2, ...t_k]$, where t_i is :

$$t_i = \frac{n_{id}}{n_d} \log(\frac{N}{n_i}) \tag{2.1}$$

where n_{id} is the number of occurrences of word i in image d, n_d is the total number of words in the image d, n_i is the number of images which contains word i and N is the total number of images in the whole database. The weighting is a product of two terms: the word frequency $\frac{n_{id}}{n_d}$, and the inverse document frequency $\log(\frac{N}{n_i})$. The intuition is that the word frequency weights words occurring often in a particular document, whilst the inverse document frequency downweights words that appear often in the database.

Video Google (Sivic and Zisserman, 2003) can be improved through different directions. (Nister and Stewenius, 2006) proposed a hierarchical TF-IDF scoring using hierarchically defined visual words that form a vocabulary tree, which allowed the use of a larger vocabulary and showed a significant improvement of retrieval. (Philbin et al., 2007) improved the efficiency by two aspects: i) it proposed to replace the direct nearest neighbor computation in K-means by an approximated K-means using kd-trees; ii) it showed that a re-ranking stage using spatial verification can efficiently improve the performance.

(Jégou et al., 2010) proposed to aggregate local descriptors through a Vector of Locally Aggregated Descriptors (VLAD), which can be seen as a simplification of the Fisher kernel (Perronnin and Dance, 2007). The idea of the VLAD descriptor is to accumulate, for each visual word μ_i, the differences $x_j - \mu_i$ of the vectors x_j assigned to μ_i. This characterizes the distribution of the vectors with respect to the center. The pseudo-code to compute VLAD is shown in Algo. 1

(Arandjelović and Zisserman, 2012) showed that three techniques can largely improve the retrieval performance: i) RootSIFT instead of SIFT (Lowe, 2004); ii) Re-ranking with discriminative query expansion; iii) Database-side feature augmentation.

Algorithm 1: Computation of the VLAD (Jégou et al., 2010) descriptor

Data: A set of descriptors $\{x_1, \ldots x_T\}$ extracted from an image. The set $\{\mu_1, \ldots \mu_K\}$ of centroids is learned on a training set using K-means. Both x_i and μ_i are d-dimenstional vectors.

Result: VLAD descriptor V $=[v_1, \ldots v_K]$, where v_i is a d-dimension vector.

Initialized v_i with 0

for $t = 1, \ldots, T$ **do**

$i = \arg\min ||x_t - \mu_j||$;

$v_i := v_i + x_t - \mu_i$

end

/* Power normalization */

for $u = 1, \ldots, Kd$ **do**

$V_u := \mathtt{sign}(V_u)|V_u|^{\alpha}$

end

/* L^2 normalization */

$V := \frac{V}{||V||_2}$

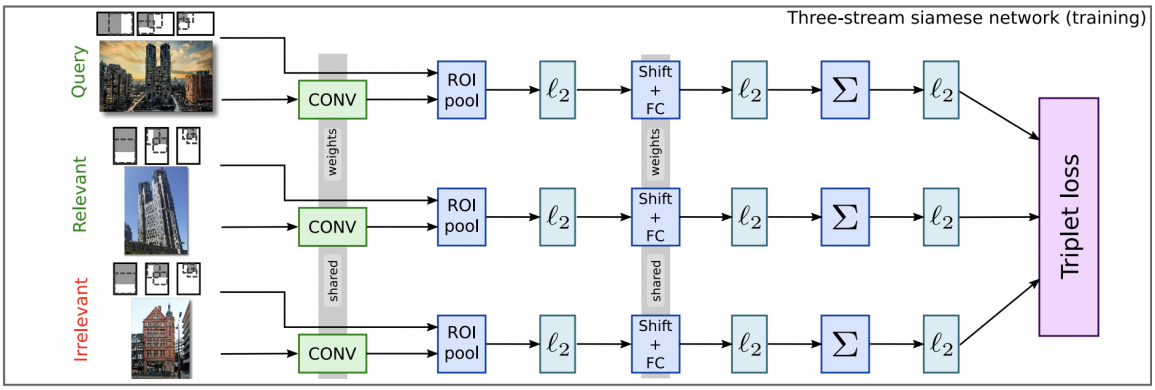


Figure 2.1: Proposed siamese network in (Gordo et al., 2017)

Deep learning based approaches. In the deep era, approaches focus on learning a good global representation of images to map similar samples closer to each other against dissimilar ones.

(Tolias et al., 2015) employed a pre-trained CNN network and extracted Regional Maximum Activation of Convolutions (R-MAC) descriptors for image retrieval. The R-MAC was an aggregation of max-pooled features of different regions at different scales. Even without any training, the R-MAC was shown to be effective on different image retrieval benchmarks.

One challenge in landmark retrieval is the lack of training set. A landmark dataset

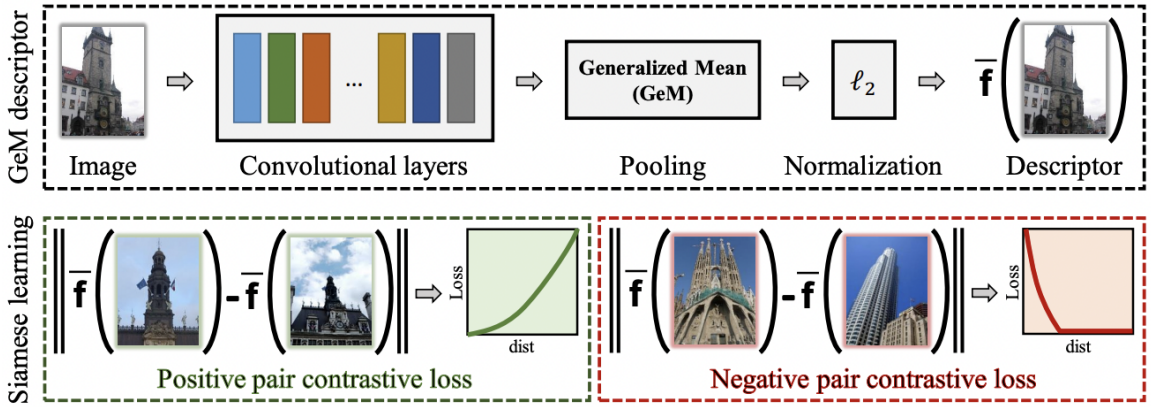


Figure 2.2: Proposed approach in (Radenović et al., 2018)

was proposed by (Babenko et al., 2014), which is a large-scale image dataset that contains approximately 214k images of 672 famous landmark sites. The images were collected through textual queries in an image search engine without thorough verification. Based on this dataset, (Gordo et al., 2017) obtained a relatively clean dataset by matching keypoints, spatial verification, and some post-processing within the images of each landmark class. In addition, (Gordo et al., 2017) also proposed to learn a siamese network based on standard triplet loss, which can be illustrated in Fig. 2.1.

(Radenović et al., 2018) leverages a structure-from-motion (SfM) pipeline (Schonberger et al., 2015) to clean the dataset. (Radenović et al., 2018) also proposed Generalized-mean pooling (GeM) to compute image-level descriptor. Let $\mathcal{X}_k$ be the set of $W \times H$ activations for feature map $k \in \{1...K\}$, where K is the total number of channels. The descriptor $\mathbf{f}^{(g)}$ after GeM is defined as:

$$\mathbf{f}^{(g)} = [f_1^{(g)}, f_2^{(g)}, ...f_K^{(g)}],\ f_k^{(g)} = (\frac{1}{|\mathcal{X}_k|}\sum_{x \in \mathcal{X}_k} x^{p_k})^{\frac{1}{p_k}} \tag{2.2}$$

where p_k is the pooling parameter, which can be manually set or learned since this operation is differentiable and can be part of the back-propagation. From a pre-trained network, (Radenović et al., 2018) conducted fine-tuning using a contrastive loss. The full pipeline is illustrated in 2.2.

(Revaud et al., 2019a) investigated a new type of ranking loss that directly optimize mAP. (Revaud et al., 2019a) leverages a histogram binning approximation, the AP can be differentiated and thus employed to end-to-end learning.

Beyond learning global pooled descriptors, NetVLAD (Arandjelovic et al., 2016)

proposed a new generalized VLAD layer inspired by the classical VLAD (Jégou et al., 2010) descriptor. The key insight is to replace the hard assignment in the clustering with a learnable soft assignment. The descriptor is defined as follows:

$$V_{j,k} = \sum_{i=1}^{N} \frac{\exp(w_k^T x_i + b_k)}{\sum_{k'} \exp(w_{k'}^T x_i + b_{k'})} (x_i(j) - \mu_k(j)) \tag{2.3}$$

where $\{w_k\}$, $\{b_k\}$ and $\{\mu_k\}$ are learnable parameters. NetVLAD (Arandjelovic et al., 2016) was trained with weak supervision, *e.g.* GPS labels, and validated on place recognition benchmarks.

To mitigate the noise with weak GPS labels, NetVLAD (Arandjelovic et al., 2016) only utilized the easiest top-1 image of the query for training, (Ge et al., 2020) proposed to improve the performance by exploring hard training samples. Recently, Patch-NetVLAD (Hausler et al., 2021) improved the performance of NetVLAD (Arandjelovic et al., 2016) through multi-scale local patches matching and geometric verification.

2.3 Object Discovery and Co-segmentation

There are a number of approaches aiming at discovering, localizing, and segmenting objects from unlabelled images. Many methods (Tang et al., 2014; Cho et al., 2015; Vo et al., 2019, 2020) use bounding box proposals and formulate the object discovery as an optimization problem. This relies on the quality of proposals that are typically not adapted for non-photorealistic data, such as artworks.

Recently, LOST (Siméoni et al., 2021) showed promising performance on object discovery using self-supervised transformer DINO (Caron et al., 2021). Their approach extracts DINO (Caron et al., 2021) features of patches and computes the similarities of the patches. The patch with the smallest number of positive correlations with other patches is considered as the initial seed for foreground regions. LOST (Siméoni et al., 2021) extends the initial seed to a region by adding patches that have positive correlations with the initial seed. The final object proposal is obtained by extracting the connected component that contains the initial seed. Although without any learning procedures, LOST (Siméoni et al., 2021) improves the state-of-the-art by a large margin.

Some approaches focus on the co-segmentation task and try to predict masks of

salient objects directly (Rother et al., 2006; Vicente et al., 2011; Rubinstein et al., 2013; Taniai et al., 2016; Yuan et al., 2017; Li et al., 2018; Hsu et al., 2018; Li et al., 2019; Chen et al., 2020b).

A representative work is (Rubinstein et al., 2013), which works well in the presence of significant amounts of noise images (images not containing a common object), as typical for datasets collected from Internet search. The main idea behind (Rubinstein et al., 2013) is that common object patterns should be salient, while being sparse with respect to smooth transformations across images. To capture the salient part in an image, it employed an off-the-shelf saliency measure proposed in (Cheng et al., 2014). Given a normalized saliency map M_i for the image i, the saliency term $\Phi_{saliency}$ at the position $\mathbf{x} = (x, y)$ is defined as:

$$\Phi^i_{saliency}(\mathbf{x}) = \log(M_i(\mathbf{x})) \tag{2.4}$$

To establish reliable correspondences between pixels in foreground regions present in different images, (Rubinstein et al., 2013) proposed a weighed SIFT objective function. Formally, let $\mathcal{F}_{i\to j}$ denote the flow field from the image i to the image j, which is computed using SIFT Flow (Liu et al., 2010). Using $\mathcal{F}_{i\to j}$ as the warping function, $\hat{\mathbf{x}}$ is the transformed position of $\mathbf{x}$, therefore $\hat{\mathbf{x}} = \mathcal{F}_{i\to j}(\mathbf{x})$. Given the binary masks $\mathbf{b_i}$, $\mathbf{b_j}$, the weighed SIFT flow objective function E_{flow} becomes:

$$E_{flow}(\mathcal{F}_{i\to j}; \mathbf{b_i}, \mathbf{b_j}) = \sum_{\mathbf{x}} \mathbf{b_i}(\mathbf{x})(\underbrace{\mathbf{b_j}(\hat{\mathbf{x}})||S_i(\mathbf{x}) - S_j(\hat{\mathbf{x}})||_1}_{\mathcal{L}_{feat}} + \underbrace{C_0(1 - \mathbf{b_j}(\hat{\mathbf{x}}))}_{\mathcal{L}_{mask}} + \underbrace{\sum_{\mathbf{y}\in\mathcal{N}^i_{\mathbf{x}}} \alpha||\mathcal{F}_{i\to j}(\mathbf{x}) - \mathcal{F}_{i\to j}(\mathbf{y})||_2)}_{\mathcal{L}_r} \tag{2.5}$$

where S_i are the dense SIFT descriptors of the image i, $\mathcal{N}^i_{\mathbf{x}}$ is the neighborhood of $\mathbf{x}$, α weighs the smoothness term, and C_0 is a large constant. The objective function contains three parts: $\mathcal{L}_{feat}$ computes dense feature similarities between two images considering displacement; $\mathcal{L}_{mask}$ pushes the estimated masks to be 1; $\mathcal{L}_r$ regularizes the flow. The difference between this objective function and the original SIFT flow (Liu et al., 2010) is that it encourages matching foreground pixels between two images.

Based on the saliency and the computed correspondences, the foreground likelihood at $\mathbf{x}$ of the image i is :

$$\Phi^i(\mathbf{x}) = \begin{cases} \Phi^i_{saliency}(\mathbf{x}) + \lambda_{match} \frac{1}{|N_i|} \sum_{j \in N_i} ||S_i(\mathbf{x}) - S_j(\hat{\mathbf{x}})|| & , \mathbf{b}_i(\mathbf{x}) = 1 \\ \beta & , \mathbf{b}_i(\mathbf{x}) = 0 \end{cases}$$

where N_i is a set of similar images for the image i computed using global descriptors. β and λ_{match} are two hyper-parameters.

Three regularization terms are considered: i) intra-image compatibility between adjacent pixels; ii) inter-image compatibility between two matched pixels; iii) color model.

For the intra-image compatibility term, it is defined as follows:

$$\Phi^i_{intra}(\mathbf{x}, \mathbf{y}) = \mathbf{1}_{\mathbf{y} \in \mathcal{N}^i_{\mathbf{x}},\, \mathbf{b}_i(\mathbf{x}) \neq \mathbf{b}_i(\mathbf{y})} \exp(-\underbrace{||I_i(\mathbf{x}) - I_i(\mathbf{y})||_2^2}_{\text{Intensity Diff.}}) \tag{2.6}$$

where $\mathbf{1}$ is the indicator function.

For the inter-image compatibility term, it is defined as follows:

$$\Phi^{i,j}_{inter}(\mathbf{x}, \hat{\mathbf{x}}) = \mathbf{1}_{\mathbf{b}_i(\mathbf{x}) \neq \mathbf{b}_j(\hat{\mathbf{x}})} \exp(-\underbrace{||S_i(\mathbf{x}) - S_j(\hat{\mathbf{x}})||_1}_{\text{Feature Diff.}}) \tag{2.7}$$

where $\hat{\mathbf{x}}$ is the corresponded pixel of $\mathbf{x}$ in the image j.

For the color model, once the masks are obtained, the color histograms of the background and foreground regions of the image i can be estimated, denoted $\mathbf{h}^0_i$ and $\mathbf{h}^0_j$. the term considering the color model is :

$$\Phi^i_{color}(\mathbf{x}) = -\log \mathbf{h}_i^{\mathbf{b}_i(\mathbf{x})}(\mathbf{x}) \tag{2.8}$$

Denote $\mathbf{B} = \{\mathbf{b}_i\}$, $\mathbf{F} = \{\mathcal{F}_{i \to j}\}$, $\mathbf{H} = \{\mathbf{h}^0_i, \mathbf{h}^1_i\}$, the final cost function is:

$$E_{total}(\mathbf{B}; \mathbf{F}, \mathbf{H}) = \sum_{i=1}^{N} \sum_{\mathbf{x}} (\Phi^i(\mathbf{x}) + \lambda_{color} \Phi^i_{color}(\mathbf{x}) + \lambda_{intra} \sum_{\mathbf{y}} \Phi^i_{intra}(\mathbf{x}, \mathbf{y}) + \lambda_{inter} \sum_{j \in N_i} \Phi^{i,j}_{inter}(\mathbf{x}, \hat{\mathbf{x}})) \tag{2.9}$$

where N_i is a set of similar images for the image i. λ_{color}, λ_{intra} and λ_{inter} are three hyper-parameters. The algorithm alternates between optimizing the correspondences $\mathbf{F}$ (Eqn. 2.5), and the binary masks $\mathbf{B}$ (Eqn. 2.9).

Some similar objective functions are also employed in the deep era for object

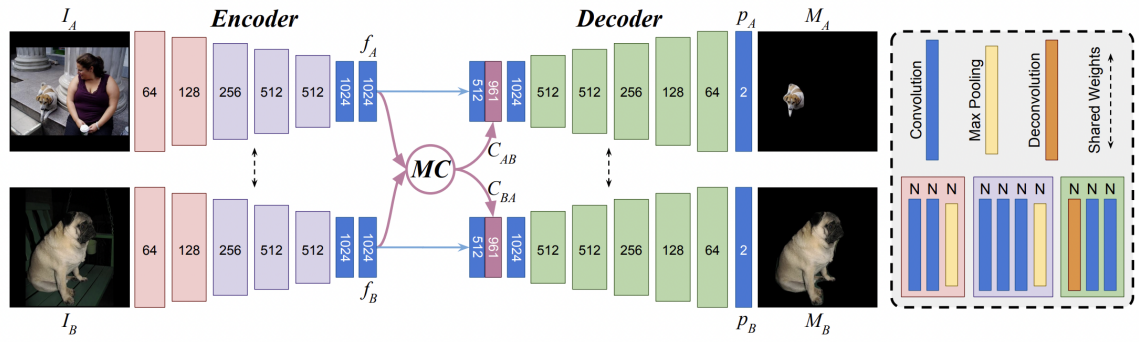


Figure 2.3: Proposed CNN architecture in (Li et al., 2018). MC is a mutual correlation layer, which is similar to Flownet (Dosovitskiy et al., 2015).

discovery. In (Chen et al., 2020b), the training losses contain five components:

$$\mathcal{L} = \mathcal{L}_{match} + \lambda_{cycle}\mathcal{L}_{cycle} + \lambda_{triplet}\mathcal{L}_{triplet} + \lambda_{contrast}\mathcal{L}_{contrast} + \lambda_{mask}\mathcal{L}_{mask} \tag{2.10}$$

where $\mathcal{L}_{match}$ is the foreground-guided matching loss which minimizes the distance between corresponding features based on the estimated geometric transformation. Both the forward-backward consistency loss $\mathcal{L}_{cycle}$ and transitivity consistency loss $\mathcal{L}_{triplet}$ regularize the network training by enforcing the predicted geometric transformations to be consistent across a pair of images ($\mathcal{L}_{cycle}$) and three images ($\mathcal{L}_{triplet}$). $\mathcal{L}_{contrast}$ optimizes the masks such that they are with higher inter-image foreground similarity and large intra-image figure-ground separation. $\mathcal{L}_{mask}$ penalizes the inconsistency of the predicted object masks of an input image pair and the estimated geometric transformations between that pair.

The solution of co-segmentation becomes much simpler if training data and annotations are available. (Li et al., 2018) proposed a pure CNN framework for object co-segmentation, which is illustrated in Fig. 2.3. MC is a mutual correlation layer, which is similar to Flownet (Dosovitskiy et al., 2015; Ilg et al., 2017). In detail, the mutual correlation layer performs a pixel-wise comparison between two feature maps f_A and f_B. Given a point (i, j) and a point (m, n) inside a patch around (i, j), the correlation between feature vectors $f_A(i, j)$ and $f_B(m, n)$ is defined as:

$$C_{AB}(i, j, k) = f_A(i, j)f_B(m, n) \tag{2.11}$$

where $k = (n - j)D + (m - i)$ and $D \times D$ is patch size. Since the common

objects can locate at any place on the two input images, the patch size is set to $D = 2\max(w-1, h-1)+1$, where w and h are the width and height of the feature maps f_A and f_B. The object co-segmentation is thus a binary image labeling problem and can be optimized using the standard cross-entropy loss function to train the network. To train the network, (Li et al., 2018) adapted the PASCAL dataset (Everingham et al., 2010) and created approximately 161K, 42K, 40K pairs for training, validation, and testing.

Another research direction is to discover discriminative patches in a dataset without using annotations. The desired patches need to satisfy two requirements: i) they need to occur frequently enough in the dataset; ii) they need to be different enough from the rest of the dataset. (Singh et al., 2012) proposed to solve the problem with an iterative algorithm.

For initialization, the input to the algorithm is a "discovery dataset“ $\mathcal{D}$ as well as a much larger "natural world dataset“ $\mathcal{N}$. Both $\mathcal{D}$ and $\mathcal{N}$ are divided into two equal, non-overlapping subsets ($\mathcal{D}_1, \mathcal{N}_1$ and $\mathcal{D}_2, \mathcal{N}_2$). S patches are randomly sampled from $\mathcal{D}_1$ at multiple scales, disallowing highly overlapping patches or patches with no gradient energy (e.g. sky patches). (Singh et al., 2012) proposed to run standard k-means clustering in HOG (Dalal and Triggs, 2005) space of the patches with k quite high to have more pure clusters. Clusters with less than 3 patches are removed.

For the iterative algorithm, given an initial set of clusters, a linear SVM classifier is trained for each cluster, using patches within the cluster as positive examples and all patches of $\mathcal{N}_1$ as negative examples. The trained discriminative classifiers are then run on the held-out validation set $\mathcal{D}_2$, and new clusters are formed from the top m firings of each detector. The new clusters are then used to train linear SVMs for the next iteration. The algorithm will be stopped if the clusters are unchanged. The final output clusters can be ordered according to their purity and discriminativeness.

As follow-up work, (Doersch et al., 2013) developed an extension of the classic mean-shift algorithm to density ratio estimation, showing that the resulting algorithm could be used for element discovery. (Doersch et al., 2013) also proposed the Purity-Coverage plot as a principled way of experimentally analyzing and evaluating different visual discovery approaches.

2.4 Dense Image Alignment

One classical approach is Lucas-Kanade (Lucas et al., 1981) is one classical approach that estimates a dense flow field between a pair of images under a brightness constancy assumption. The main limitation is that it tends to work only for small displacements. This has been partially addressed with hierarchical flow estimation (Szeliski, 2006), as well as using local features in addition to pixels to increase robustness (Brox et al., 2009; Revaud et al., 2015; Bailer et al., 2015; Hu et al., 2016). However, all such methods are still limited to aligning very similar images, where the brightness constancy assumption mostly holds.

SIFT-Flow (Liu et al., 2010) was an early method that aimed at expanding optical flow-style approaches for matching pairs of images across physically distinct, and visually different scenes. SIFT-Flow (Liu et al., 2010) proposed to extend SIFT (Lowe, 2004) descriptors to pixel-wise SIFT features S. The energy function for SIFT flow is defined as:

$$
\begin{aligned}
E_{SIFT-Flow}(\mathcal{F}_{1\to 2}) = \sum_{\mathbf{x}} & \underbrace{\min(||S_1(\mathbf{x}) - S_2(\mathcal{F}_{1\to 2}\circ\mathbf{x})||_1, t)}_{\text{Feat. Diff}} + \underbrace{\eta||\mathcal{F}_{1\to 2}\circ\mathbf{x} - \mathbf{x}||_1)}_{\text{Small Displ.}} \\
& + \underbrace{\sum_{\mathbf{y}\in\mathcal{N}^1_{\mathbf{x}}} \min(\alpha||\mathcal{F}_{1\to 2}(\mathbf{x})_u - \mathcal{F}_{1\to 2}(\mathbf{y})_u||_1, d))}_{\text{Smoothness along u}} \\
& + \underbrace{\sum_{\mathbf{y}\in\mathcal{N}^1_{\mathbf{x}}} \min(\alpha||\mathcal{F}_{i\to j}(\mathbf{x})_v - \mathcal{F}_{1\to 2}(\mathbf{y})_v||_1, d))}_{\text{Smoothness along v}}
\end{aligned}
\tag{2.12}
$$

where $\mathcal{F}_{1\to 2}$ is the flow field from image 1 to image 2, which can be also used as the warping function. $\mathbf{x}$ is the pixel position at (x, y). u and v correspond to the horizontal and vertical direction of the flow. $\mathcal{N}^1_{\mathbf{x}}$ denotes the neighborhood of $\mathbf{x}$ in the image 1. t, η, α, and d are hyper-parameters. The *Feat. Diff.* term in Eqn. 2.12 constrains the SIFT descriptors to be matched along with the flow vector. The *Small Displ.* term constrains the flow vectors to be as small as possible. The smoothness terms constrain the flow vectors of adjacent pixels to be similar.

SIFT-Flow (Liu et al., 2010) was later generalized to joint image set alignment using cycle consistency in (Zhou et al., 2015).

In the deep era, flows can be learned in a completely supervised way using synthetic data. Representative works are Flownet1 (Dosovitskiy et al., 2015) and Flownet2 (Ilg et al., 2017).

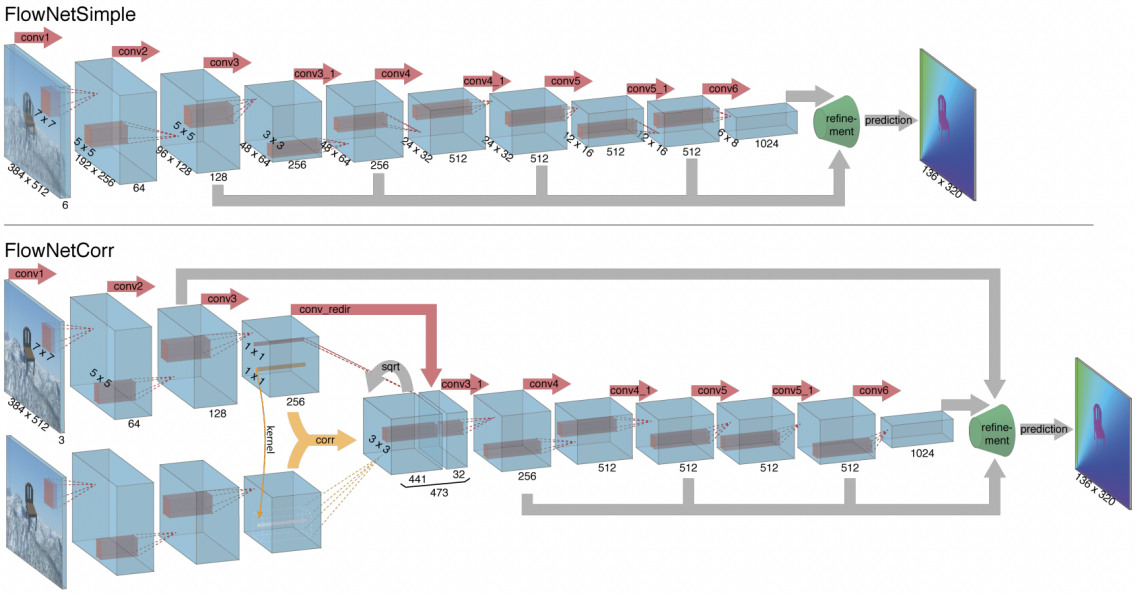


Figure 2.4: Proposed two CNN architectures in Flownet1 (Dosovitskiy et al., 2015).

In Flownet1 (Dosovitskiy et al., 2015), it proposed two architectures to learn optical flow, which are illustrated in Fig. 2.4. A simple choice is to stack the two input images together and feed them through a CNN to extract the motion information. This is referred as FlowNetSimple in Fig. 2.4 (top). Another choice is to process the correlation between features, which is the architecture FlowNetCorr in Fig. 2.4 (bottom). Formally, denote f_1 and f_2 as feature maps in image 1 and image 2. The correlation of two features at $\mathbf{x}_1$ in the first map and $\mathbf{x}_2$ in the second map is then defined as:

$$c(\mathbf{x}_1, \mathbf{x}_2) = \sum_{\mathbf{o} \in [-k,k] \times [-k,k]} f_1(\mathbf{x}_1 + \mathbf{o}) f_2(\mathbf{x}_1 + \mathbf{o}) \tag{2.13}$$

where k defines the kernel size to compare features.

To train the networks, (Dosovitskiy et al., 2015) created a simple synthetic dataset, named Flying Chairs, by applying affine transformations to images collected from Flickr and a publicly available rendered set of 3D chair models.

Flownet2 (Ilg et al., 2017) improved over Flownet1 (Dosovitskiy et al., 2015) via three points: i) a learning schedule consisting of multiple datasets; ii) a warping operation that allows stacking multiple networks; iii) a special training dataset and a specialized network for small, subpixel motion and real-world data.

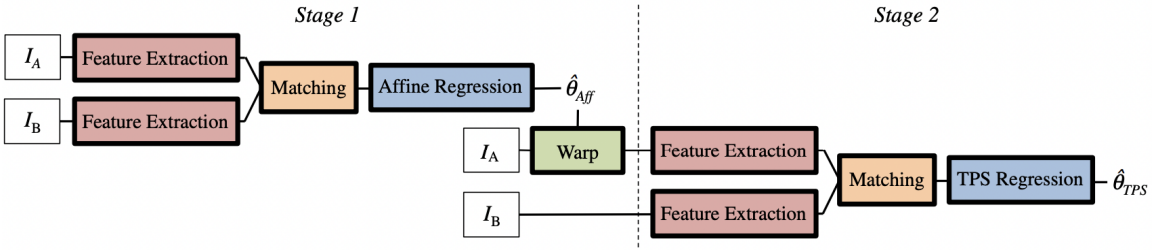


Figure 2.5: Estimating progressively more complex geometric transformations in (Rocco et al., 2017).

Generating synthetic data is also a key component to learning dense flow field between images with very different appearances. A pioneer work is (Rocco et al., 2017), which generates image pairs with randomly sampled affine and thin-plate spline (TPS) transformations. The proposed architecture is illustrated in Fig. 2.5. The network first estimates an affine transformation then a more complex TPS transformation with the warped images. Similar to Flownet1 (Dosovitskiy et al., 2015), the motion prediction is obtained through process correlations between features, which is the matching layer in Fig. 2.5. (Rocco et al., 2017) proposed measuring loss on an imaginary grid of points which is being deformed by the transformation. More precisely, a grid of points $\mathcal{G}$ in image A is transformed using the ground truth and neural network estimated transformations $\mathcal{T}_{\theta_{GT}}$ and $\mathcal{T}_{\hat{\theta}}$ with parameters θ_{GT} and $\hat{\theta}$, respectively, and measure the discrepancy between the two transformed grids:

$$\mathcal{L}(\hat{\theta}, \theta_{GT}) = ||\mathcal{T}_{\theta_{GT}} \circ \mathcal{G} - \mathcal{T}_{\hat{\theta}} \circ \mathcal{G}||_2^2 \tag{2.14}$$

Similarly, DGC-Net (Melekhov et al., 2019) learned from images pairs generated with synthetic transformation. DGC-Net adapted a coarse-to-fine strategy and included a feature correlation layer in a U-Net (Ronneberger et al., 2015) architecture to progressively improve flow resolution. The idea is illustrated in Fig. 2.6. Note that the largest resolution of the flow field predicted by DGC-Net is 240 $\times$ 240.

The follow-up work (Laskar et al., 2020) improved DGC-Net by constraining the matching layer to be locally and globally consistent and leveraging a better universal decoder.

Local and global correlation layers have also shown to be effective in GLU-Net (Prune et al., 2020). The local correlation c^l between the target $F_t^l \in \mathbb{R}^{H_l \times W_l \times d_l}$ and source

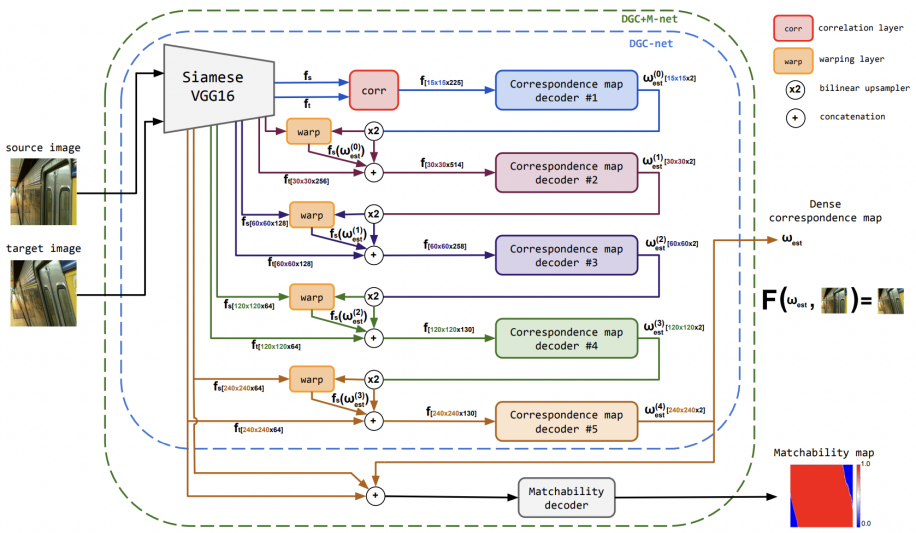


Figure 2.6: DGC-Net proposed in (Melekhov et al., 2019), which is an iterative architecture consisting of four major components: 1) the feature pyramid creator. 2) the correlation layer. 3) the fully convolutional correspondence map decoders. 4) the warping layer. The matchability decoder is a tiny CNN that predicts a confidence map of correspondences.

$F_s^l \in \mathbb{R}^{H_l \times W_l \times d_l}$ is defined as:

$$c^l(\mathbf{x}, \mathbf{d}) = F_t^l(\mathbf{x})^T F_s^l(\mathbf{x} + \mathbf{d}), \; ||d||_\infty \leq R \tag{2.15}$$

where $\mathbf{x}$ is a coordinate in the target feature map and $\mathbf{d}$ is the displacement from this location. The displacement is constrained to $||d||_\infty \leq R$, i.e. the maximum motion in any direction is R. The resulting 3D correlation volume c^l thus has a dimensionality of $H_l \times W_l \times (2R+1)^2$.

A global correlation layer evaluates the pairwise similarities between all locations in the target and source feature maps. The resulting 3D correlation volume C^l thus has a dimensionality of $H_l \times W_l \times H_l \times W_l$.

The Global-Local architecture is present in Fig. 2.7. Note that the input images have two resolutions and the global correlation is only computed for low-resolution pairs.

(Rocco et al., 2018a) fine-tuned the model from (Rocco et al., 2017) on real-world datasets with weak supervision, namely, only matched (positive) pairs are provided and no annotations of correspondences are available. The key insight is to maximize the sum of soft-inlier for positive pairs.

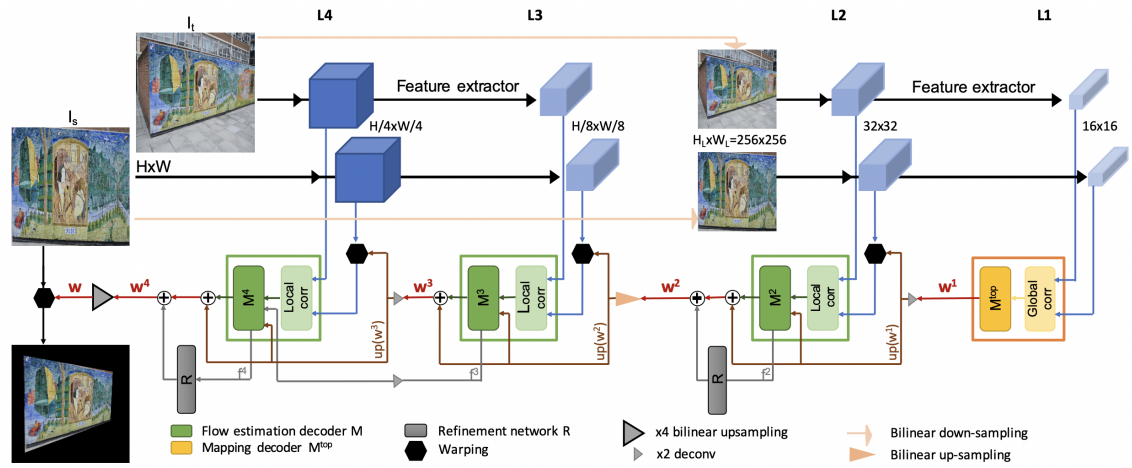


Figure 2.7: GLU-Net proposed in (Prune et al., 2020).

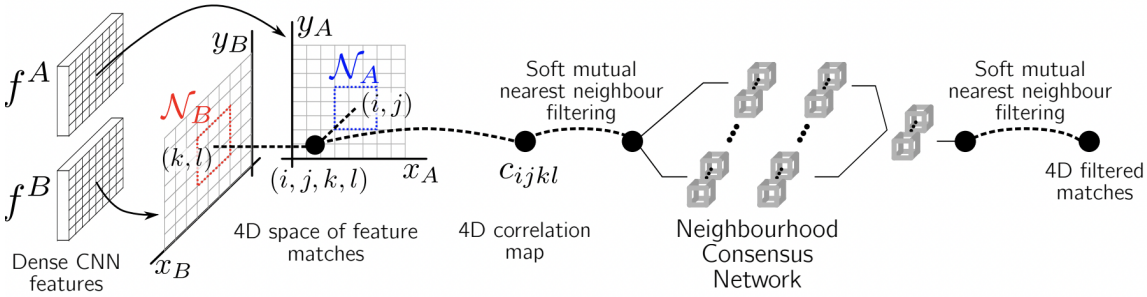


Figure 2.8: Neighbourhood Consensus Network (Nc-Net) architecture proposed in (Rocco et al., 2018b).

(Rocco et al., 2018b) introduced the idea of using 4D convolutions on the feature correlations to learn to filter neighbor consensus. The idea is illustrated in Fig. (**?**). A fully convolutional neural network is used to extract dense image descriptors f_A and f_B for images A and B, respectively. All pairs of individual feature matches $f^A_{i,j}$ and $f^B_{k,l}$ are represented in the 4D space of matches (i, j, k, l) (here shown as a 3D perspective for illustration), and their matching scores stored in the 4D correlation tensor c. These matches are further processed by 4D convolutions to produce the final set of output correspondences.

Chapter 3

Learning Features for Artworks Analysis

Discovering near duplicate patterns in large collections of artwork is harder than standard instance mining due to differences in the artistic media (oil, pastel, drawing, etc), and imperfections inherent in the copying process. The key technical insight is to adapt a standard deep feature to this task by fine-tuning it on the the specific art collection using self-supervised learning. More specifically, spatial consistency between neighbouring feature matches is used as supervisory fine-tuning signal. The adapted feature leads to more accurate style-invariant matching, and can be used with a standard discovery approach, based on geometric verification, to identify duplicate patterns in the dataset. The approach is evaluated on several different datasets and shows surprisingly good qualitative discovery results. For quantitative evaluation of the method, we annotated 273 near duplicate details in a dataset of 1587 artworks attributed to Jan Brueghel and his workshop. Beyond artwork, we also demonstrate improvement on localization on the Oxford5K photo dataset as well as on historical photograph localization on the Large Time Lags Location (LTLL) dataset. Code and data are available at http://imagine.enpc.fr/~shenx/ArtMiner

3.1 Introduction

Visiting a world-class art museum might leave one with an impression that each painting is absolutely unique and unlike any other. In reality, things are more complicated. While working on a painting, an artist would typically create a number of preliminary

(a) common detail (dog) discovered in our new Brueghel dataset of 1587 artworks

(b) relationship between painting and two studies discovered from collection of 195 artworks by Dante Gabriel Rossetti

Figure 3.1: Examples of repeated visual patterns automatically discovered by our algorithm. Sources: (a) left: *Nymphs Sleeping After the Hunt, Spied on by Satyr* (oil), center: *Diana's Nymphs After the Hunt* (oil), right: *Seventeen Studies of Different Dogs* (drawing); (b) *The Bower Meadow* (left: chalk, center: oil, right: pastel)

sketches and studies to experiment with various aspects of the composition. Many of these studies also find their way into (usually more provincial) museums. Some artists enjoy returning time and again to the same subject matter (e.g. Claude Monet and his series of 25 paintings of the same haystacks). Moreover, during the Renaissance, it was not uncommon for an artist (or an apprentice in his workshop) to reuse the same visual elements or motifs (an angel, a cart, a windmill, etc) in multiple paintings, with little or no variation. For example, Flemish painter Jan Brueghel is believed to have created many paintings that were imitations, pastiches, or reworkings of his own works, as well as these of his father, Pieter Breughel the Elder (Honig, 2016). Art historians are keenly interested in mapping out such visual connections between artworks to discover influences, find provenance, and even establish authorship. Currently, this is being done entirely by hand, with researchers spending months or even years in museum archives hoping to discover common visual patterns.

This chapter presents an approach for automatically discovering repeated visual patterns in art collections, as shown on Fig. 3.1. We propose to learn a deep visual feature able to find correspondences between near-duplicate visual elements across different artworks. This task is quite challenging, requiring a feature that is both highly discriminative (i.e. tuned to find copies of the same object instance rather then samples of an object category), but also invariant to changes in color, style, artistic media, geometric deformation, etc. Manually collecting and labelling a large enough artwork dataset containing enough variability requires enormous effort by professional art historians, which is exactly what we are trying to avoid. Therefore,

we propose a method which learns in a self-supervised way, adapting a deep feature to a given art collection without any human labelling. This is done by leveraging neighbourhood spatial consistency across matches as free supervisory signal.

Using our trained feature, we demonstrate that a simple voting and regression procedure, in line with classic verification step of instance recognition (Philbin et al., 2007), lets us discover visual patterns that are repeated across artworks within the dataset. We demonstrate our visual pattern discovery approach on several collections of artwork, including a new annotated dataset of 1587 works attributed to the Brueghel family. To further evaluate the generality of our method, we have also evaluated it on a set of historical and modern architecture photographs, resulting in state-of-the-art localization task performance on the Large Time Gap Location dataset (Fernando et al., 2015).

Our main contributions are: 1) a self-supervised approach to learn a feature for matching artistic visual content across wide range of styles, media, etc; 2) the introduction of a large new dataset for evaluating visual correspondence matching; 3) an approach to discover automatically repeated elements in artwork collections.

3.2 Related Work

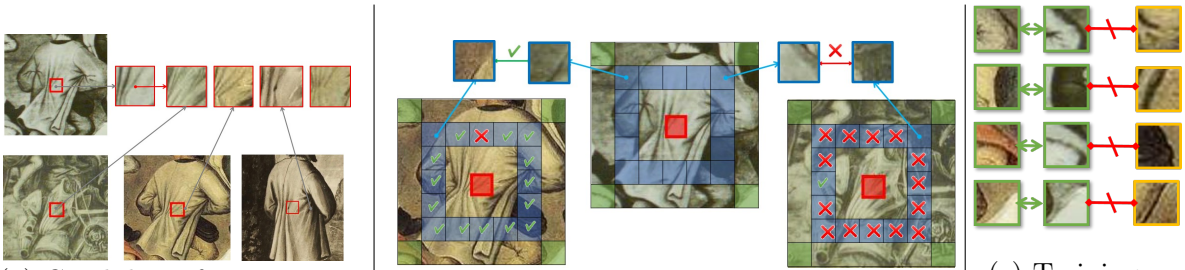
Computer vision and art. There is a long standing and fruitful collaboration between computer vision and art. On the synthesis side, promising results have been obtained for transferring artistic style to a photograph (Hertzmann et al., 2001; Gatys et al., 2016; Zhu et al., 2017), or even trying to create art (Elgammal et al., 2017; Hertzmann, 2018). On the analysis side, there are several efforts on collection and annotation of large-scale art datasets (Karayev et al., 2014; Mensink and Van Gemert, 2014; Picard et al., 2015; Wilber et al., 2017; Strezoski and Worring, 2017; Mao et al., 2017), and using them for genre and authorship classification (Karayev et al., 2014; Tan et al., 2016; Strezoski and Worring, 2017). Others focus on applying and generalizing visual correspondence and object detection methods to paintings using both classical (Shrivastava et al., 2011; Crowley and Zisserman, 2013; Aubry et al., 2014; Ginosar et al., 2014, 2018), as well as deep (Crowley et al., 2015; Crowley and Zisserman, 2016; Westlake et al., 2016; Gonthier et al., 2018). Most closely related to us is work of Yin et al. (Yin et al., 2016), which used the same Brueghel data (bru), annotating it to train detectors for five object categories (carts, cows, windmills,

rowboats and sailbaots).

Our goal, however, is to go further and focus on the computational analysis of relationships *between* individual artworks. Seguin et al. (Seguin et al., 2016, 2017) propose to find visual relationships in collections of paintings. However, while they use off-the-shelf CNNs trained in a supervised manner, we focus on the design of a new self-supervised feature learning specifically trained for the task. This allows us to focus on near-exact reproduction of detail, rather than a more generic visual similarity, which is what most art historians are actually interested in regards to specific corpora, such as the works of Brueghel family (Honig, 2016).

Spatial consistency as supervisory signal. Spatial consistency is a widely used signal in many computer vision tasks from geometry to retrieval. The classic work of Sivic et al. (Sivic and Zisserman, 2003) performs instance retrieval based on the extraction of spatially consistent local feature matches. This direction has been further developed with specially adapted features for place recognition across large visual changes (Fernando et al., 2015; Hauagge and Snavely, 2012; Verdie et al., 2015; Aubry et al., 2014). Beyond instances, this idea has been extended to discovering object categories (Cho et al., 2015) and their segmentations (Rubinstein et al., 2013). Our discovery of repeated patterns through correspondence consistency is reminiscent of the line of work on mid-level visual element discovery (Singh et al., 2012; Doersch et al., 2013, 2014). These idea have been used in the context of temporal and spatial image collection analysis, to discover the elements characteristic of a specific location (Doersch et al., 2012), or the evolution of these elements over time (Jae Lee et al., 2013).

Spatial consistency has also been used to learn deep visual features for object category in a self-supervised way, either by predicting the spatial configuration of patches (Doersch et al., 2015) or predicting the patch given its context (Pathak et al., 2016). In a similar spirit, Rocco et al. (Rocco et al., 2018b) recently demonstrated how to learn visual representations through geometric consistency to predict object-category-level correspondences between images. We, on the other hand, aim at learning features for matching only stylistically different versions of the same instance.



(a) Candidates from one proposal region

(b) Selection with verification regions

(c) Training with positive regions

Figure 3.2: Feature Learning Strategy. (a) Our approach relies on candidate correspondences obtained by matching the features of a proposal region (in red) to the full database . (b) The candidate correspondences are then verified by matching the features of the verification region (in blue) of the query in the candidate images and checking for consistency . (c) Finally, we extract features from the positive regions (in green) from the verified candidates and use them to improve the features using a metric learning loss.

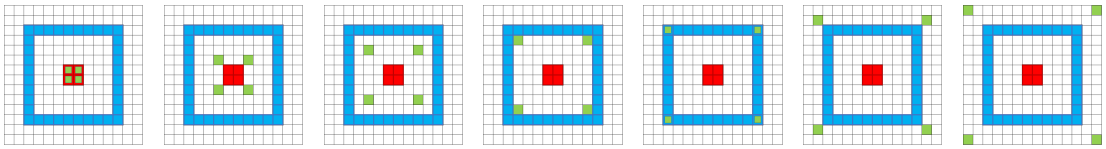
3.3 Dataset-specific Feature Learning

This section describes our strategy for adapting deep features to the task of matching artworks across styles in a specific dataset. Starting with a standard ImageNet pre-trained deep features, our idea is to extract hard-positive matching regions form the dataset that we then use in a metric learning approach to improve the features. Our two key hypothesis are that: (i) our dataset includes large parts of images that are copied from each other but are depicted with different styles, and (ii) the initial feature descriptor is good enough to extract some positive matches. Our training thus alternates between two steps that we described bellow: (1) mining for hard-positive training samples in the dataset based on the current features using spatial consistency, and (2) updating the features by performing a single gradient step on the selected samples.

3.3.1 Mining for positive feature pairs

For our approach to work, it is crucial to select positive matching examples that are both accurate and difficult. Indeed, if the features are trained with false matches, training will quickly diverge, and if the matches are too easy, no progress will be made.

To find these hard-positive matching features, we rely on the procedure visualized



P2 P4 P6 P8 P10 P12 P14

Figure 3.3: Different region configurations. red: query regions, which are used to get proposal regions, blue: verification regions, green: positive regions.

in Fig. 3.2.

Candidate sampling. *Proposal regions* are randomly sampled from each image in the dataset to be used as query features. These are matched densely at every scale to all the images in the dataset using cosine similarity in feature space. This can be done efficiently and in parallel for many queries using a normalization and a convolution layer, with the weights of the convolution defined by the query features. For each query we select one of its top K matches as candidate correspondences (Fig. 3.2a). These candidates contain a high proportion of bad matches, since most of the queries are likely not repeated K times in the dataset and since our feature is imperfect.

Candidate verification. To verify the quality of candidate matches given by the previous step, we rely on special consistency: a match will be considered valid if its neighbours agree with it. More precisely, let's assume we have a candidate match between features from the proposal region p_A in image A and a corresponding region p_B in image B, visualized in red in Fig. 3.2b. We define a *verification region* around p_A, visualized in blue. Every feature in this region is individually matched in image B, and votes for the candidate match if it matched consistently with p_B. Summing the votes of all the features in the verification region allows us to rank the candidate matches. A fixed percentage of the candidates are then considered verified. The choice of the verification region is, of course, important to the success of this verification step. The key aspect is that the features in the verification region should be, as much as possible, independent of the features in the proposal region. On the other hand, having them too far apart would reduce the chances of the region being completely matched. For our experiments, we used the 10x10 feature square centred around the query region (Fig. 3.2b).

Generating hard positives. Finally, given a set of verified correspondences, we have to decide which features to use as positive training pairs. One possibility would be to directly use features in the proposal region, since they have been verified. However, since the proposal region has already been "used" once (to verify the matches), it does not bring enough independent signal to make quality hard positives. Instead, we propose to sample positives from a different *positive region*. We evaluated different configurations for the positive region, as visualized in Fig. 3.3 (in green). We choose to keep only 4 positive pairs per verified proposal, positioned at the corners of a square and denote the different setups as P2 to P14, the number corresponding to the size of the square. We will show in the experiments that P12 and P14 perform better than the alternatives in Sec 3.5.3. The features from the positive regions (Fig. 3.2b in green) are then used as hard positives for feature fine-tuning (Fig. 3.2c).

3.3.2 Feature fine-tuning

After each selection of positive feature pairs, we update of our feature using a single gradient step of the following triplet metric learning loss:

$$\mathcal{L}(\mathcal{P}_1, \mathcal{P}_2, \{\mathcal{N}_i\}) = -\min(\lambda, s(\mathcal{P}_1, \mathcal{P}_2)) + \frac{1}{N_{neg}} \sum_{i=1}^{N_{neg}} \max(s(\mathcal{P}_1, \mathcal{N}_i), 1-\lambda) \qquad (3.1)$$

where $\mathcal{P}_1$ and $\mathcal{P}_2$ are corresponding features in the positive regions, $\{N_i\}_{\{i=1,2...N_{neg}\}}$ are negative samples, s is the cosine similarity metric and λ is a hyper-parameter. We select the negatives as the set of top matches to $\mathcal{P}_1$ in $\mathcal{P}_2$'s image. This selects hard negatives and avoids any difference in the distribution of the depiction styles in our positive and negative samples. We chose a relatively high number of negative N_{neg} to account for the fact that some of them might in fact correspond to matching regions, for example in the case of repeated elements, or for location near the optimal match.

Implementation details. In all of our experiments, we used *conv4* features of the ResNet-18 (He et al., 2016) architecture. We resized all images such that their maximum spatial dimension in the feature map was 40, leading to approximately 1k features per image at the maximum scale. For each image, we used 7 different scales, regularly sampled at two octaves with 3 scales per octave. For positive sampling, we used square queries of 2×2 features, $K = 10$ candidate matches for each query. From

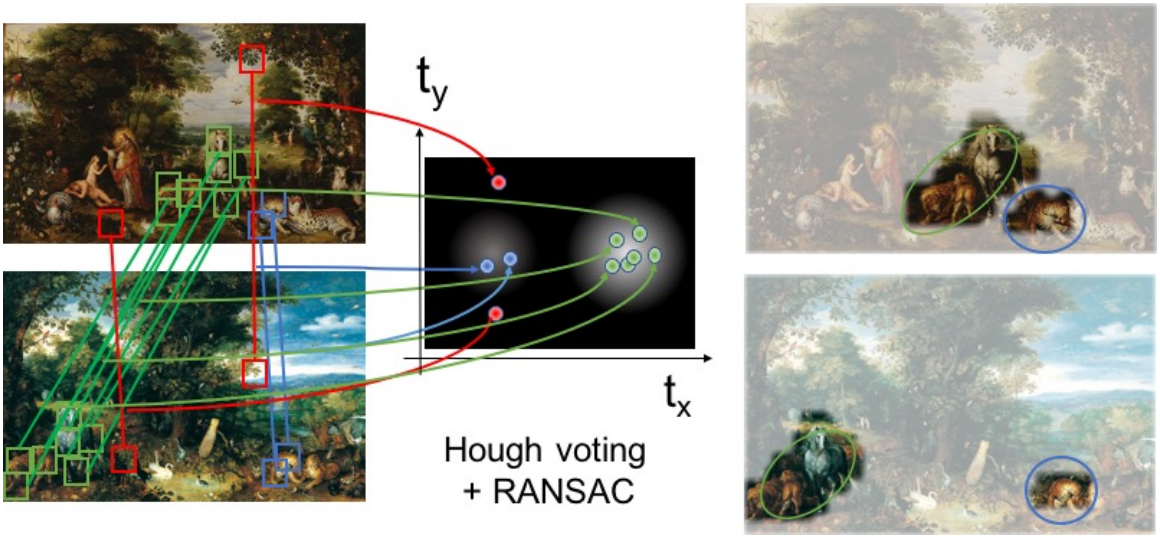


Figure 3.4: Discovery through geometric consistency.

these candidate, the top 10% with the most votes from neighbours were considered verified. Note that these parameters might need to be adjusted depending on the diversity and size of the dataset, but we found that they performed well both for the Brueghel (bru) and LTLL (Fernando et al., 2015) datasets. For training on Oxford5K (Philbin et al., 2007) dataset, the query patches are only sampled inside the annotaed bounding boxes of the 55 query images, and we only find candidate matches in 2000 images randomly sampled in the whole dataset. The hyper-parameters of the triplet loss, N_{neg} and λ, are fixed to 20 and 0.8 respectively. Our models were trained with the Adam (Kingma and Ba, 2015) optimizer with learning rate 1e-5 and $\beta = [0.9, 0.99]$. Using a single GPU Geforce GTX 1080 Ti, training converged in approximately 10 hours, corresponding to 200 iterations of the feature selection and training. Most of the time is spent extracting and verifying candidate matches. ImageNet pre-training was used for initialization in all experiments.

3.4 Spatially Consistent Pattern Mining

In this section, we describe how our algorithm discovers repeated patterns in a dataset given style-invariant features learned in the previous section. We follow the classic geometric verification approach (Philbin et al., 2007): for all pairs of images in the dataset, we first compute features and match them between the two images, then select consistent matches, and finally find image regions that have many consistent

feature matches. This allows us to build a graph between all corresponding image regions, from which we extract clusters of related images and repeated elements. In the following, we present briefly each of these steps.

3.4.1 Identifying region correspondences

Our discovery procedure for a pair of images is visualized in Fig. 3.4. We start by computing dense correspondences between the two images using our learned feature. These will be quite noisy. We first use Hough voting to identify potential groups of consistent correspondences. As a first approximation, each correspondence votes for a translation and change in scale. We then extract the top 10 translation candidates, and, using a permissive threshold, focus on the correspondences in each group independently. Within each group, we use RANSAC to recover an affine transformation and the associated inliers. This allows to account for some deformation in the copy process, but also variations in the camera viewpoint with respect to the artwork.

3.4.2 Scoring correspondences

After deformations between image regions are identified, we score the correspondence based both on the quality of the match between the features and geometric criteria. We use the following weighted average of the feature similarity:

$$S(\mathcal{I}) = \frac{1}{N} \sum_{i \in \mathcal{I}} e^{(-\frac{e_i^2}{2\sigma^2})} s_i \tag{3.2}$$

where $\mathcal{I}$ is the index of the inlier correspondences, e_i is the error between correspondence i and the geometric model, s_i the similarity of the associated descriptors and $\frac{1}{N}$ is normalization by the number of features in the source image.

3.4.3 Region correspondences graph

Using the score S, we can separate our dataset into clusters of connected images. These clusters are already interesting and visually appealing, especially for dataset with few repeated details. However, to avoid obtaining very large clusters when many details are repeated in overlapping sets of images, one needs to individually identify

each detail region. To do that we built a graph from all the connected image pairs. The nodes are the regions that are mapped between pairs of images. Each image can contain several overlapping regions. We connect regions that are matched to each other as well as regions in the same image that overlap with an Intersection over Union (IoU) score greater than a given threshold (0.5 in our experiments). Finally, we extract the connected components in this graph. Each of them corresponds to a different detail that is repeated in all images of the group.

3.5 Experiments

In this section, we analyse and evaluate our approach. We first present the main datasets we used, including our new annotations of the Brueghel dataset (bru) specifically targeted toward the new task we propose. Second, we provide detailed results and analysis for the task of one-shot visual pattern detection. Finally, we present quantitative and qualitative results for our discovery procedure.

3.5.1 Datasets

Brueghel. We introduce new annotations for the Brueghel dataset (bru), that we will release together with the images we used. Indeed, to the best of our knowledge, no other annotation for the task of near duplicate detection in artwork is currently available.

The Brueghel dataset contains 1,587 artworks done in different media (e.g. oil, ink, chalk, watercolour) and on different materials (e.g. paper, panel, copper), describing a wide variety of scenes (e.g. landscape, religious, still life) This dataset is especially adapted for our task since it assembles paintings from artists related to the same workshop, who thus had many interaction with each other, and includes many copies, preparatory drawings, and borrowed details. With the help of our art history collaborators, we selected 10 of the most commonly repeated details in the dataset and annotated the visual patterns in the full dataset using the VGG Image Annotator tool (Dutta et al., 2016). The 10 annotated patterns can be seen in Fig. 3.5 as queries (blue boxes), and our full annotation are visible in the project page `http://imagine.enpc.fr/~shenx/ArtMiner`. We were careful to select diverse patterns, and for each of them to annotate only duplicates, and not full object classes. Note for example that for the horses and lion classes, we annotated separately

Figure 3.5: Detection example with our trained features on the Brueghel dataset. We show the top 4 matches (in green) for one example of query from each of our 10 annotated categories. Notice how the matches style can be different from the one of the query.

two variants of the details (front and back facing lion, front and right facing horse). This resulted in 273 annotated instances, with a minimum of 11 and a maximum of 57 annotations per pattern.

These annotations allow us to evaluate one-shot duplicate detection results. In our evaluation, we use an IoU threshold of 0.3 for positives, because precise annotations of the bounding boxes in different environment is difficult and approximate detections would be sufficient for most applications. In practice, our detected bounding boxes, visualised in Fig. 3.5 (green boxes) often appear more consistent than the annotations. We compute the Average Precision for each query, average them per class and report class level mean Average Precision (mAP).

Large Time Lags Locations (LTLL). While our discovery algorithm targets copy detection in art, it should also be able to detect same object instances in photographs as well. We thus test our algorithm on the LTLL (Fernando et al., 2015) dataset. It contains historic and modern photographs captured from 25 locations over a time interval of more than 150 years. In total the dataset contains 225 historical and 275 modern photographs of the same locations. The task proposed in LTLL is to recognize the location of an old picture using annotated modern photographs. We use our discovery procedure to find the images most similar to the query. As in the

Table 3.1: Experimental results on Brueghel, IoU > 0.3 for both tasks.

Method (Feature)	**Cosine similarity**	**Discovery score** (Equ.3.2)
ImageNet pre-taining	58.0	54.8
Context Prediction (Doersch et al., 2015)	58.8	64.29
Ours (trained on Brueghel)	**75.3**	**76.4**
Ours (trained on LTLL)	65.2	69.95

original paper (Fernando et al., 2015), we report classification accuracy.

Oxford5K. We also evaluate our approach on Oxford buildings (Philbin et al., 2007) dataset. The dataset contains 5062 images for 11 different landmarks. We follow the standard evaluation protocol and report retrieval mAP for the 55 queries.

DocExplore. The DocExplore dataset (En et al., 2016), which is dedicated to spotting repeated patterns in manuscripts and is the closest existing dataset related to our task and provides extensive comparisons. However the repeated patterns in this dataset are rare and small, all exactly in the same style, with the same colors, and most of the data is text. We thus used it to validate our baseline one-shot detection approach, but could not use it for feature training. DocExplore contains over 1500 images with 1464 instances of 35 different details. For our experiments, we only considered the 18 largest details (the other ones corresponding to small letters).

WikiArt. To show the generality of our approach, we ran our discovery algorithm on paintings of other artists (Peter Paul Rubens, Dante Gabriel Rossetti and Canaletto) that we collected from WikiArt (wik, a,b) (respectively 387, 195 and 166 artworks).

3.5.2 One-shot detection

We evaluated our feature learning strategy using one-shot detection. This was performed simply by computing densely features on the dataset and computing their cosine similarity with the features corresponding to the query. The query was resized so its largest dimension in the feature map would be 8. Note that unlike standard deep detection approaches (Girshick et al., 2014; Ren et al., 2015), we do not use region proposals because we want to be able to match regions which do not correspond to objects.

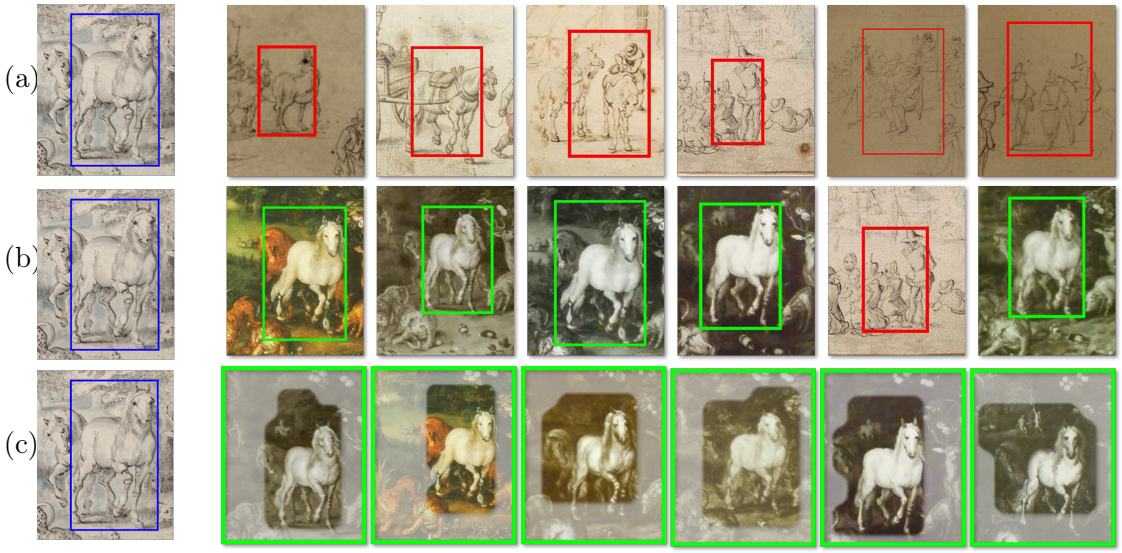


Figure 3.6: From a single query, shown on the left, we show the detection results obtained with cosine similarity with ImageNet feature (a) and our trained features (b) as well as the ones (c) obtained with our features and the discovery score presented in Sec. 3.4.2.

Examples results using this approach for each of the 10 details we annotated on the Brueghel dataset are shown in Fig. 3.5. It gives a sense of the difficulty of the task we target and the quality of the results we obtain. Note for example how the matches are of different styles, and how the two types of lions (top row) and the two types of horses (bottom row) are differentiated. In the following, we compare these results with baselines and analyse the differences.

Validation on DocExplore (doc). To validate that our one-shot detection approach is competitive with classical methods for finding repeated details, we ran it on the DocExplore (doc) dataset with ResNet-18 features trained on ImageNet. Our cosine-similarity based dense approach resulted in mAP of 55% on the 18 categories we considered, a clear improvement compared to the best performance of 40% obtained in (En et al., 2016) with classical approaches.

Comparison and analysis on Brueghel. Here, we compare the one shot detection performance with different features using cosine similarity and the score described in Eqn.3.2. In Fig. 3.6, we present the top 6 matches from the same query using different approaches. On this example, it can be seen that while ImageNet feature only gets the matches in similar styles, our trained feature obtains duplicated horses in different styles, showing that the learned feature is more invariant to style. More-

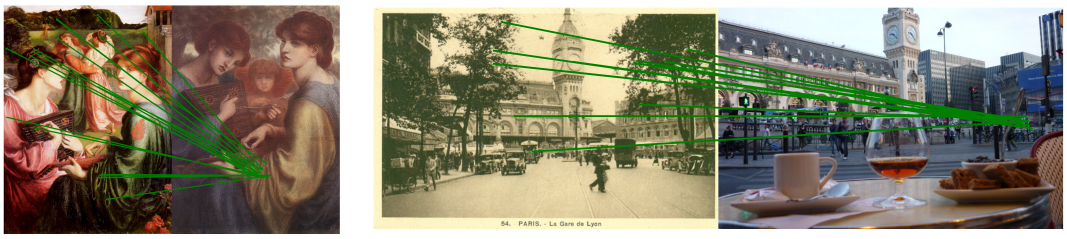


Figure 3.7: Failures of SIFT matching with geometric verification on images where our algorithm succeeds (Fig. 3.1).

over, the matching can still be improved with the discovery score. The corresponding quantitative results are presented in Tab. 3.1 and confirm these observations. Indeed learning features improves the score by approximately 30%. The discovery procedure and score provides a small additional boost, which is a first validation of our discovery procedure. We also report results for two baselines that we re-implemented: the classical Video Google (Sivic and Zisserman, 2003) approach and deep feature learnt with Context Prediction (Doersch et al., 2015). With the Video Google (Sivic and Zisserman, 2003) baseline, we obtained only 21.53% as retrieval mAP, showing the difficulty to address our task with SIFT (Lowe, 2004) features. Some failure examples are visualized in Fig. 3.7. For Context Prediction, we trained the network using the Brueghel dataset and the same ResNet18 architecture and ImageNet initialization as for our method. We only obtain an improvement of 0.8% compared to ImageNet feature, much lower than the 17% provided by our method. Interestingly, training our feature on the LTLL dataset also gave a boost in performance compared to the ImageNet feature, but is clearly worst than training on the Brueghel data, showing the dataset specific nature of our training.

3.5.3 Positive region configuration

We now focus on evaluating the different positive region settings described in Sec. 3.3.1 and Fig. 3.3. For each of them, we analyse the performance of the features on one-shot learning on the Brueghel dataset and its evolution during training. The results can be seen in Fig. 3.8. Interestingly, the performance initially always improves over ImageNet features. However, when the positive region is close to the proposal region, the performance decreases after some iterations of our training procedure, and ends up with worse performance than the initial features. But if the positive region is far enough from the query (P12 and P14), the performance improves much more and

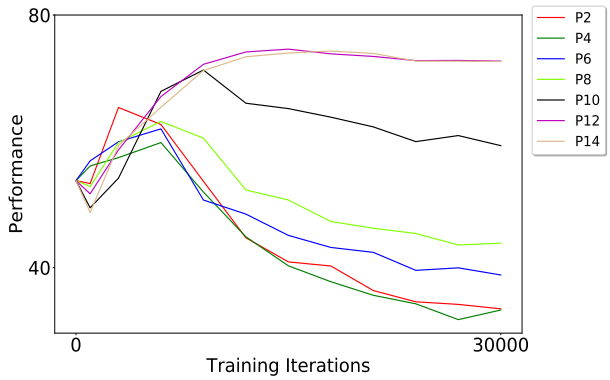


Figure 3.8: Evolution of the mean Average Precision for one-shot matching on the Brueghel dataset during training. Performance decreases after a few iterations for settings where we extract positive regions correlated with the proposal region.

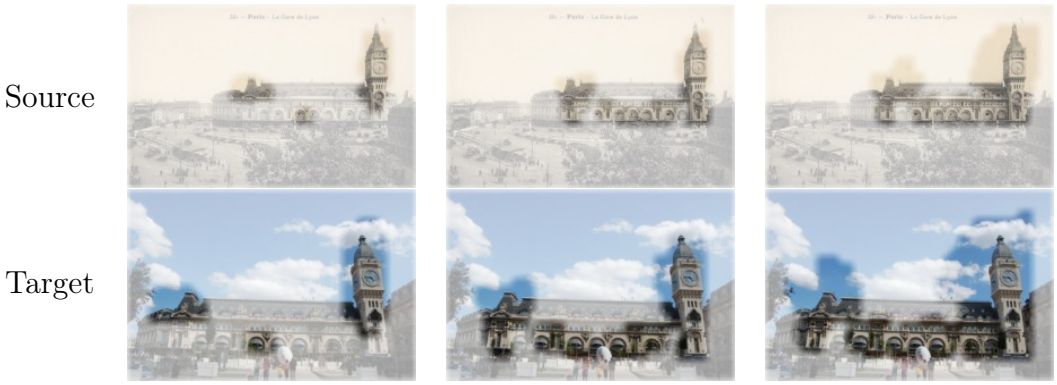


Figure 3.9: Discovery between a pair of images during training.

does not subsequently deteriorate. We thus use P12 for all our other experiments.

3.5.4 Visual pattern discovery

In this part, we focus more specifically on our discovery procedure, show qualitative results on various datasets, and evaluate quantitatively for place recognition on the LTLL dataset.

Training visualization. To visualise the influence of the feature training for our discovery task, we selected a pairs of matching images and ran discovery on them with the features at different steps of training. Fig. 3.9 visualises the results on a pair of images from the LTLL dataset. During training, larger and larger parts of the images can be matched in a consistent way, and be discovered as similar elements by

Table 3.2: Discovery with image, classification accuracy on LTLL and retrieval mAP on Oxford5K

Method	LTLL (%)	Oxford (%)
B. Fernando et al.(Fernando et al., 2015)	56.1	-
F. Radenović et al.(Radenović et al., 2018)	-	87.8
ResNet18 max-pool, image level	59.8	14.0
ResNet18 + discovery	80.9	85.0
Ours (trained LTLL + discovery)	**88.5**	83.6
Ours (trained Oxford + discovery)	85.6	**85.7**

our method. This shows both the efficiency of our feature training and it's relevance for our task.

Quantitative analysis on one-shot localisation. We evaluate our approach on one-shot localisation for both the LTLL and Oxford5K datasets. The results are reported in Tab. 3.2. We compare our discovery score to cosine similarity with standard max-pooled features as well as the state of the art results of (Fernando et al., 2015) on LTLL and (Radenović et al., 2018) on Oxford5K.

On LTLL, we use the class of the nearest neighbour in modern photographs to localise the historical images. Using the discovery score provides a very important boost compared to the results of (Fernando et al., 2015) and the max-pooled features. Using our fine-tuning procedure on the LTLL dataset improves again the results, demonstrating again the interest of our proposed dataset specific fine-tunning procedure.

Similarity, on the Oxford5K dataset, we obtain an important boost using the discovery score compared to cosine similarity with max-pooled features. Fine-tuning the features on Oxford5K improves the mAP by 0.7%. This improvement is less important than on LTLL, which is expected since there is no specific domain gap between queries and targets in the Oxford5K dataset. Our result on Oxford5K is also comparable to the state-of-the-art result obtained in (Radenović et al., 2018) which performs fine-tuning on a large image collection with ResNet101. As expected the retrieval mAP is better when fine-tuning on the Oxford dataset than on LTLL.

Qualitative analysis. We show example of our discovery results in Fig. 3.10. More results are available in our project webpage `http://imagine.enpc.fr/~shenx/ArtMiner`.

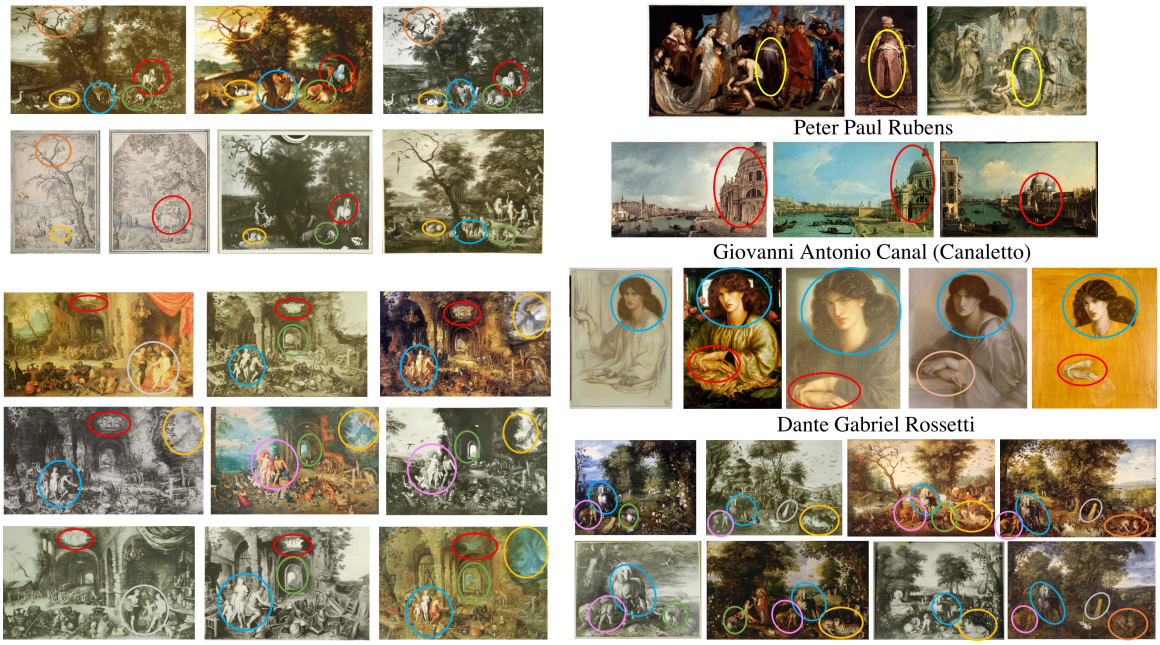


Figure 3.10: Example of image clusters discovered by our method. The three cluster without artist names correspond to data from the Brueghel dataset (bru)

Computational cost. We implement an efficient algorithm for the single shot detection by considering query features as convolutional kernels. We can thus match one query to 53 images/sec on a single GPU (Geforce GTX 1080 Ti). The discovery step is slower, since it requires matching all the features of one image to another, and takes approximately 0.2 seconds/pair of images on a single GPU. It takes about 40 minutes to query one image on Oxford5K using discovery procedure on one GPU and the discovery on the whole dataset of Brueghel took approximately 20 hours using 4 GPUs.

3.6 Limitations and Discussion

When performing discovery on different datasets, we observed some interesting failure modes visualized in Fig. 3.11. In the Brughel dataset, we discovered the identical circular frame of a set of paintings as a repeated pattern, as well as matched the faces in a set of similar but not identical paintings from portrait collections of Peter Paul Rubens.

More generally our method has several limitations. First, the time to perform

Figure 3.11: Failure examples in paintings of Peter Paul Rubens and Brughel dataset

discovery is important, which makes our approach impractical for applications such as localization. Moreover, the time for performing complete discovery on a dataset is quadratic in the number of images in this dataset, which limits the size of the datasets we can handle to a few thousands images. Second, our discovery procedure relies on an affine transformation model, which might not always be rich enough. Finally, our feature learning requires having access to a dataset which includes many repeated patterns and a good feature initialization.

3.7 Conclusion

We have introduced a new approach to adapt features for instance matching on a specific dataset without human supervision. We have demonstrated quantitatively the promise of our method both for one-shot cross-modal detection and for cross-modal instance discovery. Last but not least, we demonstrate diverse near duplicate discovery results in several artwork datasets, including some that have not been detected by humans.

Chapter 4

Learning Co-segmentation for Art Pattern Discovery

The goal of this chapter is to efficiently identify visually similar patterns from a pair of images, e.g. identifying an artwork detail copied between an engraving and an oil painting, or matching a night-time photograph with its daytime counterpart. Lack of training data is a key challenge for this task. We present a simple yet surprisingly effective approach to overcome this difficulty: we generate synthetic training pairs by taking object segments from COCO and copy-pasting them into another image. We then learn to predict the repeated object masks. We find that it is crucial to predict the correspondences as an auxiliary task and to use Poisson blending and style transfer on the training pairs to generalize on real data. We analyse results with two deep architectures relevant to our joint image analysis task: a transformer-based (Vaswani et al., 2017) architecture and Sparse Nc-Net (Rocco et al., 2020), a recent network designed to predict coarse correspondences using 4D convolutions. We show our approach provides clear improvements for artwork details retrieval on the Brueghel dataset (bru; Shen et al., 2019) and achieves competitive performance on two place recognition benchmarks, Tokyo247 (Torii et al., 2015) and Pitts30K (Torii et al., 2013). We then demonstrate the potential of our approach by performing object discovery on the Internet object discovery dataset (Rubinstein et al., 2013) and the Brueghel dataset (bru; Shen et al., 2019).

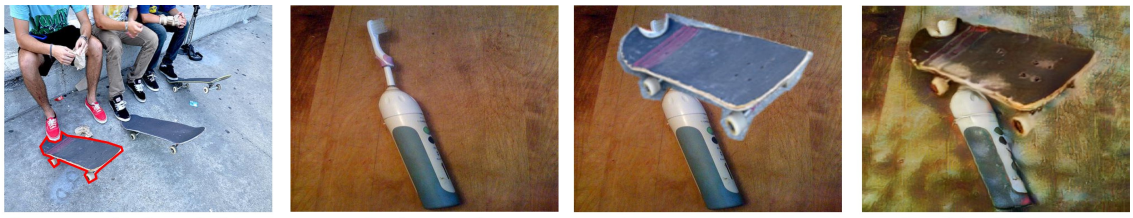
4.1 Introduction

Identifying repeated patterns lies at the very heart of the computer vision problem, and is a key component of Intelligence itself. Yet, in practice, our best methods for performing such a fundamental task often leave a lot to be desired. E.g. while we now have good methods for discovering *exact* pattern matches (used extensively to find copyright infringements), as well as approximate matches of *salient* objects (see object discovery and co-segmentation approaches in Section 2), detecting visually similar details within a larger visual context remains surprisingly difficult. Spotting the repetition of visual detail has several applications. E.g. identifying copied details in artworks allows art historians to discover influences, find provenance, and establish authorship (Shen et al., 2019). Matching repeated details can boost performances in visual localisation for place recognition (Hausler et al., 2021).

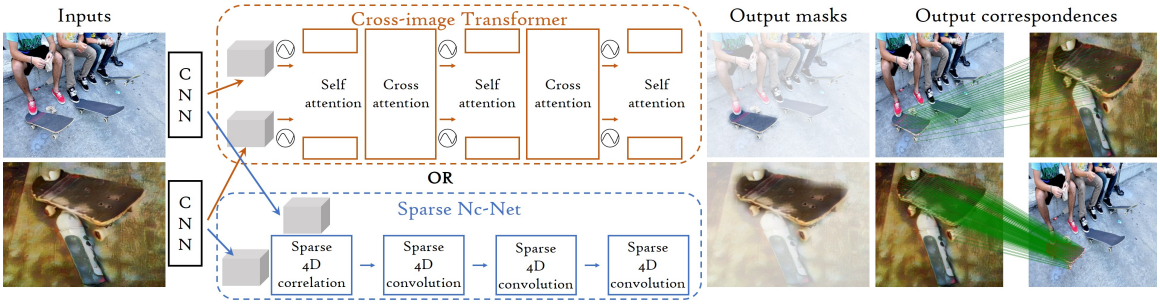
Reliable pair-wise image co-segmentation and correspondence identification could also enable object discovery in image collections (Chen et al., 2020b). However, identifying repeated content in image pairs remains challenging, especially in the cases where images appear very different from each other. Moreover, there is no available generic training dataset for this task.

In this chapter, we show it is possible to learn to detect repeated visual patterns – jointly predicting co-segmentation and correspondences – without any human-labelled correspondences. Instead, we generate synthetic correspondence pairs via automatic data augmentation. More precisely, we use a "segment swapping" approach, where we blend object segments in a random background using Poisson blending and apply style transfer to the resulting image to obtain challenging training image pairs (Fig. 4.1a). We compare using as image segments either COCO (Lin et al., 2014) instance segmentation or unsupervised segments, which produced slightly lower but comparable results. On the generated image pairs, we have access to the ground-truth matchability masks as well as the correspondences which we use as supervisions for training a network (Fig. 4.1b). Surprisingly, we find that models trained on such a dataset generalize well to real data. We experimented with two network architectures which we adapt to predict co-segmentation and correspondences in image pairs: the recent Sparse Nc-Net (Rocco et al., 2020) architecture, designed for predicting image coarse correspondences, and an architecture based on Transformers (Vaswani et al., 2017) which we refer to as cross-image transformer.

We analyze the effectiveness of our data generation process, architectures and



(a) Data generation by "segment swapping". Instead of directly pasting an object from a source image on a background (3rd column), we use Poisson blending (Pérez et al., 2003) and add style transfer (Huang and Belongie, 2017) to the result (4th column).



(b) We train our cross-image transformer or Sparse Nc-Net (Rocco et al., 2020) on the generated pairs. Both networks jointly predict masks and correspondences.

Figure 4.1: Learning co-segmentation by "segment swapping". We generate training data with "segment swapping" (Fig. 4.1a) and learn co-segmentation either with our cross-image transformer or Sparse Nc-Net (Rocco et al., 2020) (Fig. 4.1b).

training strategy on two types of tasks. First, we perform retrieval tasks using the predicted pair-wise co-segementation masks and correspondences. We show clear performance improvement for artwork details retrieval on the Brueghel (Shen et al., 2019) dataset and results comparable to state of the art for visual localization on two challenging place recognition benchmarks, Tokyo247 (Torii et al., 2015) and Pitts30K (Torii et al., 2013). This last result is especially impressive, since these benchmarks are very competitive, and many dedicated methods leveraging geo-referenced images or real correspondence for supervision have been proposed. On the contrary, our approach is generic and relies solely on our synthetic "segment swapping" training. We further make use of the predicted masks and correspondences to build a candidate correspondence graph and perform discovery with spectral clustering (Ng et al., 2001; Leordeanu and Hebert, 2005). We demonstrate results on par with state-of-the-art on the standard co-segmentation Internet (Rubinstein et al., 2013) dataset and show qualitative results on the challenging Brueghel (Shen et al., 2019) dataset.

Our full code is available at our project page `https://imagine.enpc.fr/`

~shenx/SegSwap/.

4.2 Related Work

Learning correspondences between different images. SIFT-Flow (Liu et al., 2010) was an early method that aligns visually distinct scenes by incorporating visual features, such as SIFT (Lowe, 2004), into optical flow style approaches. More recently, many deep learning based approaches have been developed to predict correspondences from correlations of input features (Rocco et al., 2017; Melekhov et al., 2019; Prune et al., 2020; Shen et al., 2020a; Truong et al., 2021b,a). Of particular interest, architectures based on attention mechanisms and Transformers (Vaswani et al., 2017) have been introduced to predict image correspondences. SuperGlue (Sarlin et al., 2020) is an attention-based graph neural network for key-point matching. Closer to this chapter, COTR (Jiang et al., 2021) is an sequence-to-sequence transformer architecture that takes an image and 2D coordinates of a query points as inputs to predict correspondences. Finally, LoFTR (Sun et al., 2021) adopts a coarse-to-fine approach to matching with a transformer encoder. As opposed to our work, these transformer-based methods are trained on a large dataset with ground-truth poses and depth while we only train on a synthetic dataset. Additionally, our model is only composed of an encoder and outputs a mask of the common regions along with the correspondences.

Learning correspondences without annotated data. There is a large body of work that use synthetic images (Dosovitskiy et al., 2015) or images with synthetic deformations (Rocco et al., 2017; Seo et al., 2018; Melekhov et al., 2019; Prune et al., 2020) to learn correspondences without real annotated training data. However, these approaches do not try to identify the matchable regions, which are essential to discover visual details. Some other approaches train directly on real images using proxy signals for correspondences, such as photometric or cycle consistency (Zhou et al., 2015; Wang et al., 2018b; Janai et al., 2018; Shen et al., 2020a; Truong et al., 2021b,a). Again, they focus on the quality of the correspondences and are not designed to predict matchable regions in vastly different images. On the contrary, the core of our approach is to discover these similar regions. This makes our approach particularly suited for retrieval tasks.

Our approach is also inspired by related data augmentation techniques, specifically, the CopyPaste augmentation used by Ghiasi et al. (Ghiasi et al., 2021) for instance segmentation and the stylised-ImageNet augmentation used in Geirhos et al. (Geirhos et al., 2019) to increase shape bias in neural networks.

Object discovery and co-segmentation. There is a wide variety of approaches aiming at discovering objects and their location from unlabelled images. Many methods (Tang et al., 2014; Cho et al., 2015; Vo et al., 2019, 2020) use bounding box proposals and formulate the object discovery as an optimization problem. This relies on the quality of proposals which are typically not adapted for non-photorealistic data, such as artworks. Other approaches (Rother et al., 2006; Vicente et al., 2011; Rubinstein et al., 2013; Taniai et al., 2016; Yuan et al., 2017; Li et al., 2018; Hsu et al., 2018; Li et al., 2019; Chen et al., 2020b) focus on predicting masks of salient objects directly. Some (Yuan et al., 2017; Li et al., 2018, 2019) require foreground masks for training, while others (Joulin et al., 2010, 2012; Vicente et al., 2011; Hsu et al., 2018; Li et al., 2019; Chen et al., 2020b) are designed to segment common repeated objects in a image collection. These approaches make strong assumptions about the frequency of appearance of an object, while, in many practical scenarios, common objects are rare and discovering them is about seeking a needle in a haystack (bru; Shen et al., 2019). Our approach is related to (Rubinstein et al., 2013; Taniai et al., 2016), as we both leverage dense correspondences to discover objects. As opposed to our work, Taniai et al. (Taniai et al., 2016) focuses on a single pair of images while we also show results over an entire collection of images. Rubinstein et al. (Rubinstein et al., 2013) makes the assumption that the common object is also the most salient in the image. This works well with images form internet queries but does not apply to artworks where the common object can be a detail in a richer scene.

4.3 Co-segmentation by Segment Swapping

We show an overview of our approach in Fig. 4.1. In Sec. 4.3.1, we introduce our "segment swapping" data generation process (Fig. 4.1a). We then present in Section 4.3.2 the two architectures we use (Fig. 4.1). We discuss our loss and training strategy in Section 4.3.3.

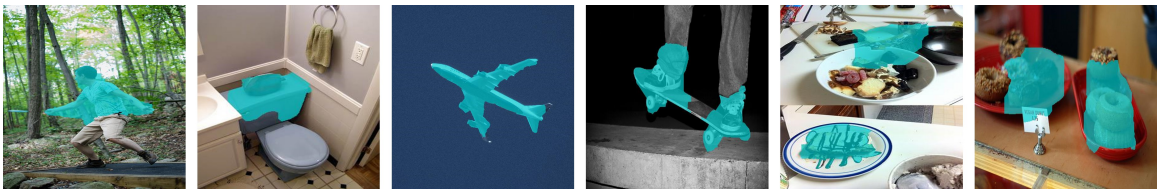


Figure 4.2: Segments extracted without any human annotations

4.3.1 Training data generation by segment swapping

We generate training pairs using the COCO dataset (Lin et al., 2014). We first sample a source image, from which we extract one or two objects. We then build the target image by applying geometric transformations to the objects and blending them into a random background image using Poisson blending (Pérez et al., 2003). The geometric transformations include rotation, translation, scaling, and thin-plate spline (TPS). A style augmentation is then performed on both the source and target images using an AdaIN (Huang and Belongie, 2017) model trained on the Brueghel dataset (bru; Shen et al., 2019). We provide more examples of training samples in the project page.

Training pairs generation. We generate training pairs using images from the COCO dataset (Lin et al., 2014). We first sample a source image, from which we extract one or two segments (as explained below). We then build the target image by applying geometric transformations to the objects and blending them into a random background image using Poisson blending (Pérez et al., 2003). The geometric transformations include rotation, translation, scaling, and thin-plate spline (TPS). A style augmentation is then performed on both the source and target images using an AdaIN (Huang and Belongie, 2017) model trained on the Brueghel dataset (bru; Shen et al., 2019). An example of training pair can be seen in Fig. 4.1a and we provide more examples of training samples in the project page `https://imagine.enpc.fr/~shenx/SegSwap/`.

Segments definition. The simplest way to define segments for our data generation process is to use the instance annotations from COCO (Lin et al., 2014). We compare this approach to a completely unsupervised segment extraction, which we defined using the following strategy:

(1) given an image, we compute bounding box proposals via the Selective Search (Uijlings et al., 2013);

(2) we compute a simple saliency for each feature in the bounding box, i.e. for each Moco (Chen et al., 2020a) *Conv4* feature (i,j) inside the box we use as saliency its average cosine distance with its neighbours, which is illustrated in Equ. 4.1.

$$Saliency(i,j) = \frac{1}{|\mathcal{N}(i,j)|} \sum_{(m,n)\in\mathcal{N}(i,j)} 1 - \cos(\mathbf{F}(i,j), \mathbf{F}(m,n)) \tag{4.1}$$

where $\mathcal{N}(i,j)$ is the set containing neighbours of (i, j), cos denotes the cosine similarity and $\mathbf{F}(i,j)$ is the *Conv4* feature at position (i, j). We only keep boxes which average saliency is high enough (≥ 0.5);

(3) the final mask for each box is obtained using GrabCut (Rother et al., 2004) initialized by the saliency map.

The examples of unsupervised segments are provided in Fig. 4.2.

4.3.2 Architectures

Our networks take as input a source image $\mathbf{I}^s$ and a target image $\mathbf{I}^t$, from which features maps $\mathbf{F}^s$ and $\mathbf{F}^t$ of spatial dimension $W \times H$ are extracted by a feature extraction backbone. These feature maps are then processed either by our cross-image transformer or our sparse Nc-Net (Rocco et al., 2020) architecture to predict both the masks of the repeated objects in the source and target images, $\mathbf{M}^s \in [0,1]^{W\times H}$ and $\mathbf{M}^t \in [0,1]^{W\times H}$ respectively, and the correspondences both from source to target $\mathbf{C}^{s\to t}$ and target to source $\mathbf{C}^{t\to s}$. Both $\mathbf{C}^{s\to t}$ and $\mathbf{C}^{t\to s}$ are represented as matrices of size $W \times H \times 2$. To simplify notation, we sometime use the masks as continuous 2D functions, which in practice is done by performing bilinear interpolation.

Cross-image transformer. We built an architecture based on the classic transformer encoder (Vaswani et al., 2017) which alternates multi-headed attention and fully connected feed-forward networks (FFN) blocks. The FFN blocks contain two layers with a ReLu non-linearity. Similar to (Sarlin et al., 2020), we use two types of attention layers: one is the standard self-attention (SA) layer, the other one is a cross attention (CA) layer where the attention is only computed between features from different images. We include the same 2D positional encoding as DeTR (Carion et al., 2020) on top of the feature map before SA. Our transformer alternates these two types of attention layers as shown in Fig. 4.1b, with a total of five attention and FFN blocks. Each attention layer has 2 heads and the dimension of the features is 256. Our last layer is followed by a sigmoid and has three outputs, that we interpret

as masks and correspondences for each image. We provide an ablation study of this architecture in the project page.

Sparse Nc-Net. Nc-Net (Rocco et al., 2020) is designed to learn coarse correspondences under weak supervision. It takes as input the correlations between $\mathbf{F}^s$ and $\mathbf{F}^t$, seen as a 4D volume of affinities $\mathcal{A}_{input} \in \mathbb{R}^{W\times H\times W\times H}$, and processes them with 4D convolutions. The final 4D convolution predicts affinities $\mathcal{A}_{pred} \in \mathbb{R}^{W\times H\times W\times H}$, on which softmax functions are applied in dimensions corresponding to source and target giving $\mathcal{A}^s_{pred}(i,j,k,l) = \frac{\exp(\mathcal{A}_{pred}(i,j,k,l))}{\sum_{k,l}\exp(\mathcal{A}_{pred}(i,j,k,l))}$ and $\mathcal{A}^t_{pred}(i,j,k,l) = \frac{\exp(\mathcal{A}_{pred}(i,j,k,l))}{\sum_{i,j}\exp(\mathcal{A}_{pred}(i,j,k,l))}$. We use the maxima of these affinities as source and target masks, i.e., $\mathbf{M}^s(i,j) = \max_{k,l}\mathcal{A}^s_{pred}(i,j,k,l)$ and $\mathbf{M}^t(k,l) = \max_{i,j}\mathcal{A}^t_{pred}(i,j,k,l)$. Correspondences are obtained with soft-argmax:

$$\begin{aligned}\mathbf{C}^s(i,j) &= \left(\sum_{k,l}\frac{k}{W}\mathcal{A}^s_{pred}(i,j,k,l), \sum_{k,l}\frac{l}{H}\mathcal{A}^s_{pred}(i,j,k,l)\right)\\ \mathbf{C}^t(k,l) &= \left(\sum_{i,j}\frac{i}{W}\mathcal{A}^t_{pred}(i,j,k,l), \sum_{i,j}\frac{j}{H}\mathcal{A}^t_{pred}(i,j,k,l)\right).\end{aligned} \tag{4.2}$$

Since 4D convolutions are computational heavy, we instead use sparse 4D convolutions with the same architecture as Sparse Nc-Net (Rocco et al., 2020).

4.3.3 Loss and training

On our synthetic training data we have access to the ground truth masks $\mathbf{M}^s_{gt}$ and $\mathbf{M}^t_{gt}$ and ground truth correspondences $\mathbf{C}^{s\to t}_{gt}$ and $\mathbf{C}^{t\to s}_{gt}$ on the source and target images. Our loss is the sum of two symmetric terms for source and target, for simplicity we write only the source loss $\mathcal{L}^s_{sup}$. It includes a cross-entropy (CE) loss on the predicted mask $\mathcal{L}_{mask}$ and the transported mask $\mathcal{L}_{tmask}$, as well as a regression loss $\mathcal{L}_{corr}$ on the correspondences:

$$\mathcal{L}^s_{sup} = \underbrace{CE(\mathbf{M}^s_{gt}, \mathbf{M}^s)}_{\mathcal{L}_{mask}} + \underbrace{CE(\mathbf{M}^s_{gt}, \mathbf{M}^t(\mathbf{C}^{s\to t}))}_{\mathcal{L}_{tmask}} + \underbrace{\eta\frac{1}{\sum_{i,j}\mathbf{M}^s_{gt}(i,j)}\sum_{i,j}\mathbf{M}^s_{gt}(i,j)\|\mathbf{C}^{s\to t}(i,j) - \mathbf{C}^{s\to t}_{gt}(i,j)\|}_{\mathcal{L}_{corr}} \tag{4.3}$$

where i and j correspond to the feature coordinates, η is a scalar hyper-parameter, and $CE(\mathbf{M}_{gt}, \mathbf{M}) = -\frac{1}{W\times H}\sum_{i,j}\mathbf{M}_{gt}(i,j)\log(\mathbf{M}(i,j)) + (1-\mathbf{M}_{gt}(i,j))\log(1-\mathbf{M}(i,j))$. Note that this loss is computed both for positive pairs (source and target pairs generated

by segment swapping) and negative pairs (sampled from two different pairs, without repeated objects) for which $\mathbf{M}^s_{gt} = \mathbf{M}^t_{gt} = 0$ and by convention $\mathcal{L}_{corr} = 0$.

Implementation details. We implement our approach using the Pytorch library and our full implementation is available in the project page `https://imagine.enpc.fr/~shenx/SegSwap/`. We use as backbone features the *conv4* features of a ResNet-50 (He et al., 2016) trained on ImageNet (Deng et al., 2009) with MOCO-v2 (Chen et al., 2020a). We freeze the backbone during the training, as learning backbone features leads to overfiting on the synthetic training set. For all the experiments, we optimise the loss defined in Eqn. 4.3 with $\eta = 8$ and use the Adam optimiser (Kingma and Ba, 2015) with momentum terms $\beta_1 = 0.5$ and $\beta_2 = 0.999$. At each iteration, we sample 5 positive and 15 negative pairs. For the transformer architecture, after training 200k iterations with learning rate 2e-4, we train with hard negative pairs and learning rate 1e-5 for 5k iterations. Hard negatives are obtained by sampling a pool of $N_{pool} = 500$ images from different synthetic pairs, computing predicted masks for all the pairs of images in the pool, and keeping those with mask prediction higher than a threshold $\tau = 0.04$ in a hard negative pair pool for $K_{hard} = 1000$ iterations of training. For Sparse Nc-Net (Rocco et al., 2020) training 200k iterations with learning rate 2e-4 without hard negative mining leads to the best performance. The entire trainings of the transformer and Sparse Nc-Net (Rocco et al., 2020) take approximately 30 hours and 15 hours respectively on a single GPU Tesla-V100-16GB. An ablation study of the architectures and more training details are provided in the project page.

4.4 Application to Image Retrieval

In this section, we show how our model can be used for retrieval tasks. We first explain how we use it to compute an image similarity score in Sec. 4.4.1. We then present experimental results in Sec. 4.4.2, including art detail retrieval on the Brueghel dataset (bru; Shen et al., 2019) and place recognition on Pitts30k (Torii et al., 2013) and Tokyo 24/7 (Torii et al., 2015).

Table 4.1: Art detail retrieval and detection on Brueghel (Shen et al., 2019; bru). For detection, we employ ArtMiner (Brueghel (Shen et al., 2019) + cos (Shen et al., 2019)) as a post-processing and reports results with IoU > 0.3 (Shen et al., 2019)

Feat. + Methods	mAP	
	Retrieval	Det.(IoU > 0.3)
Brueghel (Shen et al., 2019) + cos (Shen et al., 2019)	75.5	75.3
Brueghel (Shen et al., 2019) + discovery (Shen et al., 2019)	76.6	76.4
MocoV2 (Chen et al., 2020a) + cos (Shen et al., 2019)	79.0	78.7
MocoV2 (Chen et al., 2020a) + discovery (Shen et al., 2019)	80.8	79.6
Ours + Unsupervised segments		
Transformer	83.3	79.8
Sparse-Ncnet	82.4	73.5
Ours + COCO segments (Lin et al., 2014)		
Transformer	**84.4**	**81.8**
Sparse-Ncnet	83.3	73.7

4.4.1 Score between a pair of images

We propose the following score $\mathcal{S}$ to measure the similarity between a pair of images based on predicted correspondences and masks. $\mathcal{S}$ is the sum of weighted local features similarities, where our predicted correspondences are used to associate features and the weight $\mathbf{M}^s_{joint}$ is the product of the source and transported target mask $\mathbf{M}^s_{joint}(i,j) = \mathbf{M}^t(\mathbf{C}^{s\to t}(i,j))\mathbf{M}^s(i,j)$:

$$\mathcal{S}(\mathbf{I}^s, \mathbf{I}^t) = \sum_{i,j} \underbrace{\mathbf{M}^s_{joint}(i,j)}_{\text{Mask}} \underbrace{cos(\mathbf{F}^s(i,j), \mathbf{F}^t(\mathbf{C}^{s\to t}(i,j)))}_{\text{Feat. similarity}} \tag{4.4}$$

4.4.2 Experiments

Qualitative results on our different datasets can be seen in Fig. 4.3. The predicted masks, shown with transparency, are able to capture repeated regions even in challenging cases, such as large difference of scale, viewpoints, lightening conditions and depiction styles. More visual results are provided in the project page https://imagine.enpc.fr/~shenx/SegSwap/.

Art detail retrieval. We evaluate our approach on the Brueghel dataset (bru; Shen et al., 2019) in Tab. 4.1. Our score allows us to directly retrieve images from a selected query detail. To further compare with the detection performance in ArtMiner (Shen et al., 2019), we crop a 320 × 320 patch around the predicted regions

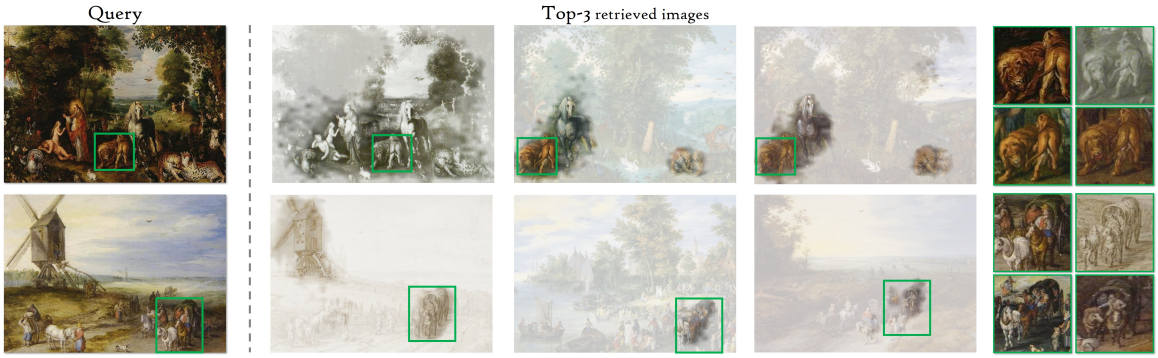
Table 4.2: Image-based localization on Tokyo 24/7 (Torii et al., 2015) and Pitts30k (Torii et al., 2013). We follow Patch-NetVLAD (Hausler et al., 2021) and re-rank the top-100 images ranked by NetVLAD (Arandjelovic et al., 2016) features.

Method	Supervision	Tokyo 24/7			Pitts30k-test		
		R@1	R@5	R@10	R@1	R@5	R@10
AP-GEM (Revaud et al., 2019a; Hausler et al., 2021)	Image location	40.3	55.6	65.4	75.3	89.3	92.5
DenseVLAD (Torii et al., 2015; Hausler et al., 2021)	Image location	59.4	67.3	72.1	77.7	88.3	91.6
NetVLAD (Arandjelovic et al., 2016; Hausler et al., 2021)	Image location	73.3	82.9	86.0	86.0	93.2	95.1
CRN (Kim et al., 2017a; Ge et al., 2020)	Image location	75.2	83.8	87.3	-	-	-
SARE (Liu et al., 2019; Ge et al., 2020)	Image location	79.7	86.7	90.5	-	-	-
IBL (Ge et al., 2020)	Image location	**85.4**	**91.1**	**93.3**	-	-	-
Re-ranking Top-100 from NetVLAD (Arandjelovic et al., 2016; Hausler et al., 2021)							
Patch-NetVLAD (Hausler et al., 2021)	Image location	81.9	85.7	87.9	88.6	94.5	95.8
Patch-NetVLAD (Hausler et al., 2021) + RANSAC	Image location	86.0	88.6	**90.5**	**88.7**	94.5	95.9
SuperGlue (Sarlin et al., 2020; Hausler et al., 2021)*	Pose+Depth	**88.2**	**90.2**	90.2	**88.7**	**95.1**	**96.4**
Ours + Unsupervised segments							
Transformer	Segment swapping	76.5	82.9	85.4	83.5	92.9	95.3
Nc-Net	Segment swapping	83.2	87.0	87.6	85.6	94.1	95.5
Ours + COCO segments (Lin et al., 2014)							
Transformer	Segment swapping	80.0	86.0	87.9	84.7	93.5	95.6
Nc-Net	Segment swapping	85.4	88.3	89.2	86.8	94.4	95.8

* uses learnt keypoint detector Superpoint (DeTone et al., 2018)

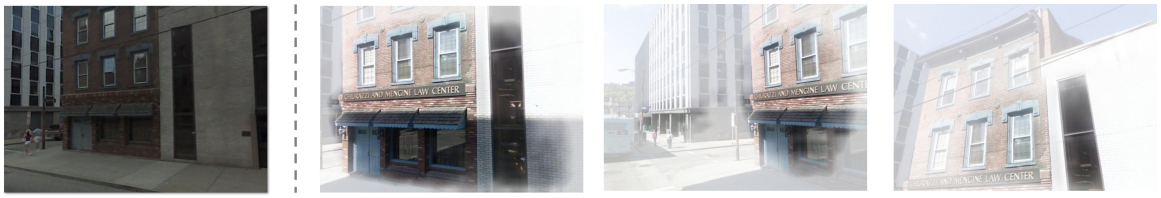
and use ArtMiner (Shen et al., 2019) as a post-processing to obtain the bounding box prediction. The correspondences are more accurate for the cross-image transformer which achieves much better results for detection. We also observe that, in this benchmark, the performances with unsupervised segments are close to the ones using COCO (Lin et al., 2014) instance annotations, which suggests that our approach does not depend on human annotations. Note that the best performance of ArtMiner is obtained with a discovery score which is expensive to compute and involves multi-scale feature matching and RANSAC. Our approach is thus simpler, faster and more effective.

Place recognition. In Tab. 4.2 we compare our approach to state of the art for place recognition on the Pitts30k (Torii et al., 2013) and Tokyo 24/7 (Torii et al., 2015) datasets. The descriptions of the datasets are in the project page. We follow the standard evaluation protocol (Sattler et al., 2012; Gronat et al., 2013; Torii et al., 2013; Arandjelović and Zisserman, 2014; Torii et al., 2015; Ge et al., 2020). The query image is correctly localized if one of the top N retrieved database images is within d = 25 meters from the ground truth TUM coordinate of the query. The recall is then reported for N = 1, 5, 10. For Tokyo 24/7, we follow (Torii et al., 2015; Ge et al., 2020) and perform spatial non-maximal suppression on ranked database images before evaluation. To enable fast evaluation, we follow PatchVlad (Hausler et al., 2021) and



(a) Retrieval results on Brueghel (bru; Shen et al., 2019). green bounding-boxes are one-shot detection results.

(b) Retrieval results on Tokyo24/7 (Torii et al., 2015).



(c) Retrieval results on Pitts30k (Torii et al., 2013).

Figure 4.3: Visual results for retrieval on different datasets. For each query image (1st column), we show its 3 most similar images with the predicted masks as transparency. For Brueghel (bru; Shen et al., 2019), we also show the detection results.

evaluate our score on the top-100 images given by NetVLAD (Arandjelovic et al., 2016). Although our approach is not specifically designed for place recognition, it achieves performances comparable to Patch-NetVLAD (Hausler et al., 2021) without RANSAC. Note that the competing approaches either employ specific supervisions or more complicated process such as RANSAC, while our approach is trained only with our synthetic segment swapping data. Note that on this task where retrieving discriminative repeated regions is sufficient and correspondence accuracy is not critical, the Nc-Net architecture preforms better. Similar to the Brueghel results, leveraging COCO (Lin et al., 2014) annotated segments leads to superior performance. Training with unsupervised segments still leads to competitive results using the NC-Net archi-

Table 4.3: Ablation study. We report retrieval mAP on Brueghel (bru; Shen et al., 2019) and R@1 on Tokyo 24/7 (Torii et al., 2015) with our cross-image transformer using COCO segments (Lin et al., 2014).

Dataset		Losses			Cross-image Transformer	
Posson blending	Style transfer	$\mathcal{L}_{mask}$	$\mathcal{L}_{tmask}$	$\mathcal{L}_{corr}$	Brueghel mAP	Tokyo 24/7 R@1
✓	✓	✓	✓	✓	**84.4**	**80.0**
✗	✓	✓	✓	✓	75.1	60.0
✓	✗	✓	✓	✓	75.6	57.8
✓	✓	✗	✓	✓	80.9	67.8
✓	✓	✓	✗	✓	79.8	61.3
✓	✓	✓	✓	✗	8.5	13.3

tecture. However, it gives clearly worst results using the transformer architecture on Tokyo 24/7. We think this performance gap could be bridged using more advanced unsupervised segments.

Ablation study. An ablation study of our approach using the cross-image transformer architecture is shown in Tab. 4.3 on the Brueghel (bru; Shen et al., 2019) and Tokyo24/7 (Torii et al., 2015) datasets. We notice that: *(i)* Poisson blending (Pérez et al., 2003) and style transfer (Huang and Belongie, 2017) are both critical; *(ii)* the three terms of the loss are necessary for good performance. More analysis on the importance of learning correspondences for the generalization of the mask predicition, the similarity score and the architectures are provided in the project page `https://imagine.enpc.fr/~shenx/SegSwap/`.

4.5 Application to Object Discovery and Co-segmentation

4.5.1 Correspondences graph and clustering

In the spirit of (Leordeanu and Hebert, 2005), we see object discovery as a graph clustering problem, where the vertices $\mathcal{V}$ of the graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ are correspondences between images and the weights of the edges encodes consistency between the correspondences. Let us consider a set of N images $(I_1, \ ..., \ I_N)$. For every pair of images our network predicts correspondences that we add to the set of vertices $\mathcal{V}$ if the associated mask value is higher than a threshold. Each vertex $v_i = (s_i, t_i, x_i^s, x_i^t, m_i)$

Table 4.4: Co-segmentation on Internet (Rubinstein et al., 2013). We report pixel level precision $\mathcal{P}$ and Jaccard index $\mathcal{J}$

Method	Airplane		Car		Horse		Avg	
	$\mathcal{P}$	$\mathcal{J}$	$\mathcal{P}$	$\mathcal{J}$	$\mathcal{P}$	$\mathcal{J}$	$\mathcal{P}$	$\mathcal{J}$
DOCS (Li et al., 2018)*	0.946	0.64	0.940	0.83	0.914	0.65	0.933	0.70
Sun et al. (Sun and Ponce, 2016)	0.886	0.36	0.870	0.73	0.876	0.55	0.877	0.55
Joulin et al. (Joulin et al., 2010)	0.493	0.15	0.587	0.37	0.638	0.30	0.572	0.27
Kim et al. (Kim et al., 2011)	0.802	0.08	0.689	0.00	0.751	0.06	0.754	0.05
Rubinstein et al. (Rubinstein et al., 2013)	0.880	0.56	0.854	0.64	0.828	0.52	0.827	0.43
Chen et al. (Chen et al., 2014)	0.902	0.40	0.876	0.65	0.893	0.58	0.890	0.54
Quan et al. (Quan et al., 2016)	0.910	0.56	0.885	0.67	0.893	0.58	0.896	0.60
Hati et al. (Hati et al., 2016)	0.777	0.33	0.621	0.43	0.738	0.20	0.712	0.32
Chang et al. (Chang and Wang, 2015)	0.726	0.27	0.759	0.36	0.797	0.36	0.761	0.33
Lee et al. (Lee et al., 2015)	0.528	0.36	0.647	0.42	0.701	0.39	0.625	0.39
Jerripothula et al. (Jerripothula et al., 2016)	0.905	0.61	0.880	0.71	0.883	0.61	0.889	0.64
Jerripothula et al. (Jerripothula et al., 2017)	0.818	0.48	0.847	0.69	0.813	0.50	0.826	0.56
Hsu et al. (Hsu et al., 2018)	0.936	0.66	0.914	0.79	0.876	0.59	0.909	0.68
Chen et al. (Chen et al., 2020b)	0.941	0.65	**0.940**	0.82	**0.922**	**0.63**	**0.935**	**0.70**
Ours + Unsupervised segments								
transformer	**0.941**	0.66	0.919	0.79	0.887	0.57	0.916	0.67
Nc-Net	0.682	0.19	0.791	0.56	0.774	0.27	0.749	0.34
Ours + COCO segments (Lin et al., 2014)								
transformer	**0.941**	**0.67**	0.928	**0.82**	0.916	0.60	0.928	**0.70**
Nc-Net	0.655	0.23	0.857	0.61	0.873	0.43	0.795	0.42

* learned with strong supervision (i.e., manually annotated object masks)

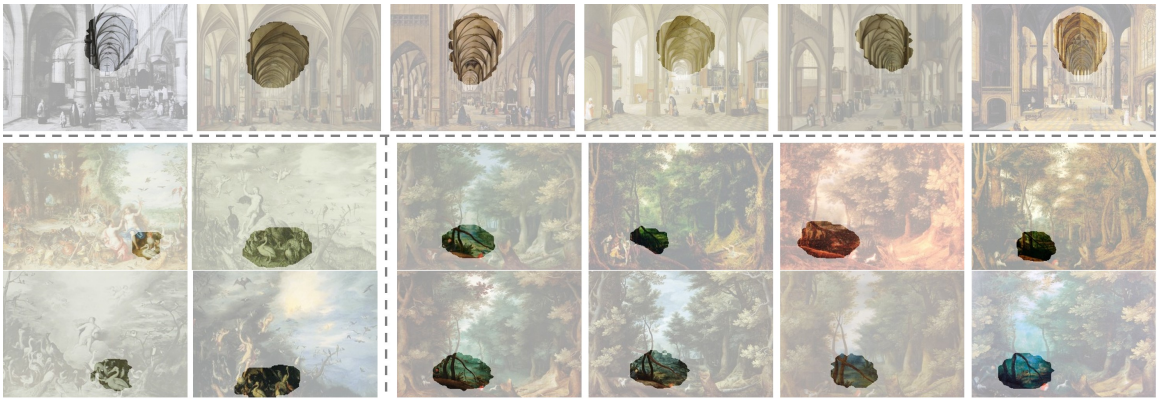
in the graph is thus associated to a predicted correspondence and defined by the indices s_i and t_i of the images it connects, the associated coordinates x_i^s and x_i^t and the predicted mask value m_i. We use cycle consistency to define the weights of the edges between the different vertices. More precisely, we only connect correspondences which have exactly one image in common. For example, let's assume that we have two vertices v_i and v_j such that If $s_i = s_j = s$ and $t_i \neq t_j$. We use our network to predict correspondence fields $\mathbf{C}^{t_i \to t_j}$ and $\mathbf{C}^{t_j \to t_i}$ and we define the weight $\epsilon_{i,j}$ of the edge between v_i and v_j as:

$$\epsilon_{i,j} = \frac{1}{2} m_i m_j exp(-\frac{\|x_i^s - x_j^s\|}{\sigma}) \left(exp(-\frac{\|x_i^t - \mathbf{C}^{t_j \to t_i}(x_j^t)\|}{\sigma}) + exp(-\frac{\|x_j^t - \mathbf{C}^{t_i \to t_j}(x_i^t)\|}{\sigma}) \right) \quad (4.5)$$

where σ is a scalar hyper-parameter. The edges are defined similarly in the cases $s_i = t_j$, $t_i = s_j$ and $t_i = t_j$. More details about the way we define the graph and in particular strategies to limit the number of vertices are given in the project page.

Given the correspondence graph, we use the spectral decomposition of its adjacency matrix (Ng et al., 2001; Leordeanu and Hebert, 2005) either to obtain clusters of correspondences for object discovery, or a foreground potential for co-segmentation.

(a) Co-segmentation results in Internet (Rubinstein et al., 2013) dataset for the Horse, Airplane and Car categories.



(b) Examples of discovered clustered on Brueghel (bru; Shen et al., 2019).

Figure 4.4: Visual results for object discovery. We show co-segmentation results on the Internet (Rubinstein et al., 2013) in Fig. 4.4a and some discovered clusters in Brueghel (Shen et al., 2019; bru) in Fig. 4.4b.

For discovery we first compute N_{eig} principal eigenvectors then performing K-means with $K_{cluster}$ clusters. For co-segmentation, we directly use the first eigenvector to define a foreground potential. Note that because we only consider in the graph correspondences with mask values higher than a threshold, the full graph is extremely sparse that the eigen-decomposition can be efficiently computed.

4.5.2 Experiments

Object co-segmentation on Internet dataset (Rubinstein et al., 2013). We build the correspondences graph using for each image only the correspondences in the five most similar images according to the retrieval score of Eqn. 3.2. We then use the principal eigen-vector of the correspondence graph to define a seed for Grab-

Cut (Rother et al., 2004). More precisely, for every image we associate to each position the sum of the eigen-vector values for the correspondences at this position. Note that GrabCut (Rother et al., 2004) is crucial to achieve good performance on this dataset, and is widely used by competing approaches such as (Rubinstein et al., 2013; Jerripothula et al., 2016; Quan et al., 2016; Hsu et al., 2018; Chen et al., 2020b). More details about the GrabCut (Rother et al., 2004) can be found in the project page. We follow the standard evaluation protocol (Rubinstein et al., 2013; Chen et al., 2020b) and report pixel-level precision $\mathcal{P}$ and the Jaccard index $\mathcal{J}$ on three subsets: Airplane, Car, Horse. The precision $\mathcal{P}$ measures pixel accuracy. The Jaccard index $\mathcal{J}$ is the IoU between the segmented object and ground truth object. Quantitative results are presented in Tab. 4.4 and qualitative results in Fig. 4.4b. Our cross-image transformer obtains performance comparable to the state of the art unsupervised approaches. Again, the performances using annotated COCO (Lin et al., 2014) segments and unsupervised segments are close, which demonstrates that the success of our approach does not come from implicitly leveraging annotated object segmentations. Sparse Nc-Net performances are clearly worse for this task. This can be understood by looking at qualitative results: the segmentation masks predicted by Nc-Net tend to be more localized in discriminative regions.

Discovery on Brueghel dataset (bru; Shen et al., 2019). Images are resized to 640 × 640, as many repeated details in Brueghel (bru; Shen et al., 2019) are small. We also remove duplicate images and images with similar borders to focus on more interesting repeated details.

Again, we only include in the graph the correspondences from the five most similar images according to the retrieval score to limit the size of the graph and we perform K-means for $K_{cluster} = 500$ clusters with $N_{eig} = 100$ principal eigen vectors. The graph has ∼900K nodes and it took 10 hours to compute predictions of all the pairs and 2 hours to perform the eigen-decomposition and clustering. Fig. 4.4b presents some interesting clusters that are not covered by ArtMiner (Shen et al., 2019)[1]. More results and details are in the project page `https://imagine.enpc.fr/~shenx/SegSwap/`.

[1] http://imagine.enpc.fr/~shenx/ArtMiner/visualRes/brueghel/brueghel.html

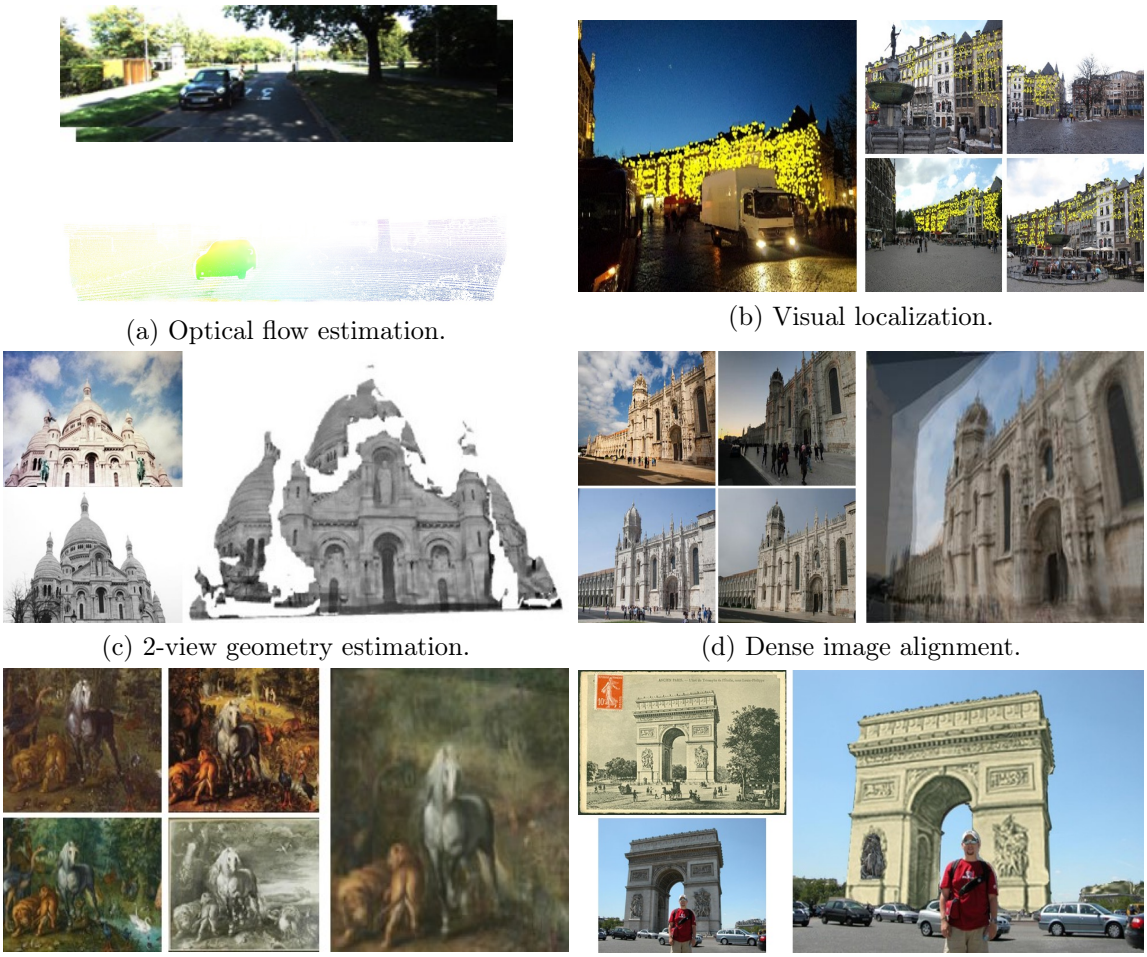
4.6 Conclusion

In this chapter, we presented a "segment swapping" approach to generate pairs of images with repeated patterns from which we show it is possible to train co-segementation and correspondence prediction networks. We evaluated two architectures, a cross-image transformer and Sparse Nc-Net (Rocco et al., 2020). We also compared using annotated segments in COCO (Lin et al., 2014) and segments extracted in a completely unsupervised way, which shows that our approach is not reliant on COCO (Lin et al., 2014) object annotations. The trained models shows competitive or better performance on various datasets and different tasks, including art detail retrieval, place recognition and object discovery.

Chapter 5

Unsupervised Dense Image Alignment

In the chapter, we consider the generic problem of dense alignment between two images, whether they be two frames of a video, two widely different views of a scene, two paintings depicting similar content, etc. Whereas each such task is typically addressed with a domain-specific solution, we show that a simple unsupervised approach performs surprisingly well across a range of tasks. Our main insight is that parametric and non-parametric alignment methods have complementary strengths. We propose a two-stage process: first, a feature-based parametric coarse alignment using one or more homographies, followed by non-parametric fine pixel-wise alignment. Coarse alignment is performed using RANSAC on off-the-shelf deep features. Fine alignment is learned in an unsupervised way by a deep network which optimizes a standard structural similarity metric (SSIM) between the two images, plus cycle-consistency. Despite its simplicity, our method shows competitive results on a range of tasks and datasets, including unsupervised optical flow on KITTI, dense correspondences on HPATCHES, two-view geometry estimation on YFCC100M, localization on AACHEN DAY-NIGHT, and, for the first time, fine alignment of artworks on the BRUGHEL DATASET. Our code and data are available at http://imagine.enpc.fr/~shenx/RANSAC-Flow/.



(a) Optical flow estimation.

(b) Visual localization.

(c) 2-view geometry estimation.

(d) Dense image alignment.

(e) Artwork alignment.

(f) Texture transfer.

Figure 5.1: RANSAC-Flow provides competitive results on a wide variety of tasks and enables new challenging applications.

5.1 Introduction

Dense image alignment (also known as image registration) is one of the fundamental vision problems underlying many standard tasks from panorama stitching to optical flow. Classic work on image alignment can be broadly placed into two camps: parametric and non-parametric. Parametric methods assume that the two images are related by a global parametric transformation (e.g. affine, homography, etc), and use robust approaches, like RANSAC, to estimate this transformation. Non-parametric methods do not make any assumptions on the type of transformation, and attempt to directly optimize some pixel agreement metric (e.g. brightness constancy constraint in

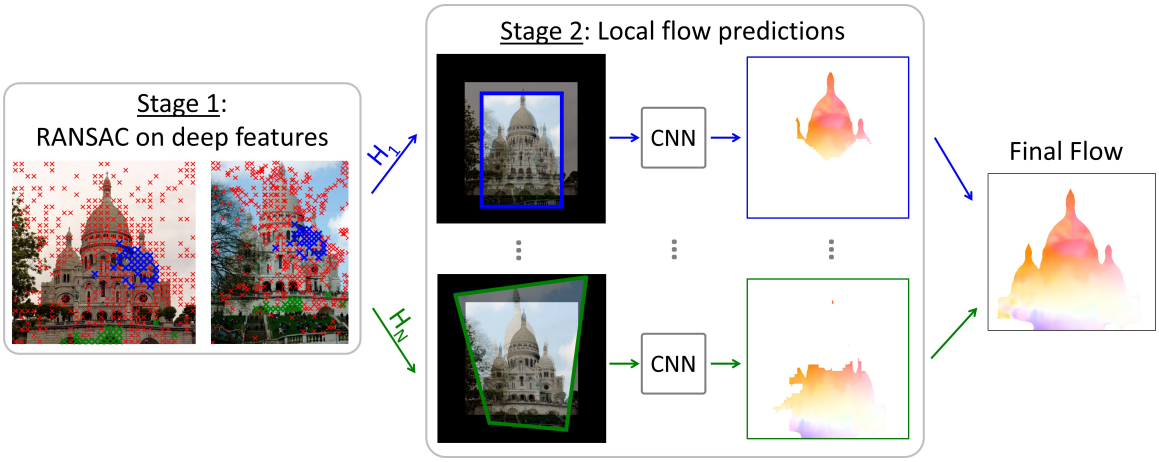


Figure 5.2: **Overview of RANSAC-Flow.** Stage 1: given a pair of images, we compute sparse correspondences (using off-the-shelf deep features), use RANSAC to estimate a homography, and warp second image using it. Stage 2: given two coarsely aligned images, our self-supervised fine flow network generates flow predictions in the matchable region. To compute further homographies, we can remove matched correspondences, and iterate the process.

optical flow and stereo). However, both approaches have flaws: parametric methods fail (albeit gracefully) if the parametric model is only an approximation for the true transform, while non-parametric methods have trouble dealing with large displacements and large appearance changes (e.g. two photos taken at different times from different views). It is natural, therefore, to consider a hybrid approach, combining the benefits of parametric and non-parametric methods together.

In this chapter, we propose RANSAC-flow, a two-stage approach integrating parametric and non-parametric methods for generic dense image alignment. Fig. 5.2 shows an overview. In the first stage, a classic geometry-verification method (RANSAC) is applied to a set of feature correspondences to obtain one or more candidate coarse alignments. Our method is agnostic to the particular choice of transformation(s) and features, but we've found that using multiple homographies and off-the-shelf self-supervised deep features works quite well. In the second non-parametric stage, we refine the alignment by predicting a dense flow field for each of the candidate coarse transformations. This is achieved by self-supervised training of a deep network to optimize a standard structural similarity metric (SSIM) (Wang et al., 2004) between the pixels of the warped and the original images, plus a cycle-consistency loss (Zhou et al., 2016).

Despite its simplicity, the proposed approach turns out to be surprisingly effective. The coarse alignment stage takes care of large-scale viewpoint and appearance variations and, thanks to multiple homographies, is able to capture a piecewise-planar approximation of the scene structure. The learned local flow estimation stage is able to refine the alignment to the pixel level without relying on the brightness constancy assumption. As a result, our method produces competitive results across a wide range of different image alignment tasks, as shown in Fig. 5.1: (a) unsupervised optical flow estimation on KITTI (Menze and Geiger, 2015a) and HPATCHES (Balntas et al., 2017), (b) visual localization on AACHEN DAY-NIGHT (Sattler et al., 2018), (c) 2-view geometry estimation on YFCC100M (Thomee et al., 2016), (d) dense image alignment, and applications to (e) detail alignment in artwork and (f) texture transfer. Our code and data are available at `http://imagine.enpc.fr/~shenx/RANSAC-Flow/`.

5.2 Related Work

Feature-based image alignment. The classic approach to align images with very different appearances is to use sparse local image features, such as SIFT (Lowe, 2004), which are designed to deal with large viewpoint and illumination differences as well as clutter and occlusion. These features have to be used together with a geometric regularization step to discard false matches. This is typically done using RANSAC (Fischler and Bolles, 1981; Raguram et al., 2012; Barath and Matas, 2018; Barath et al., 2019) to fit a simple geometric transformation (e.g. affine or homography) (Szeliski, 2006). Recently, many works proposed to learn better local features (Luo et al., 2019; DeTone et al., 2018; Tian et al., 2017; Mishchuk et al., 2017; Luo et al., 2018; Revaud et al., 2019b). Differentiable and trainable version of RANSAC have also been developed (Zhang et al., 2019; Qi et al., 2017a; Plötz and Roth, 2018; Ranftl and Koltun, 2018).

Using mid-level features (Singh et al., 2012; Kim et al., 2017b, 2018, 2019) instead of local keypoints, proved to be beneficial for matching visual content across modalities, e.g. 3D models and paintings (Aubry et al., 2014). Recently, (Shen et al., 2019) learned deep mid-level features for matching across different visual media (drawings, oil paintings, frescoes, sketches, etc), and used them together with spatial verification to discover copied details in a dataset of thousands of artworks. (Rocco et al., 2017) used deep feature map correlations as input to a regression network on syn-

thetic image deformations to predict the parameters of an affine or thin-plate spline deformation. Finally, transformer networks (Jaderberg et al., 2015) can also learn parametric alignment typically as a by-product of optimizing a classification task.

Direct image alignment. Direct, or pixel-based, alignment has its roots in classic optical flow methods, such as Lucas-Kanade (Lucas et al., 1981), who solve for a dense flow field between a pair of images under a brightness constancy assumption. The main drawback is these methods tend to work only for very small displacements. This has been partially addressed with hierarchical flow estimation (Szeliski, 2006), as well as using local features in addition to pixels to increase robustness (Brox et al., 2009; Revaud et al., 2015; Bailer et al., 2015; Hu et al., 2016). However, all such methods are still limited to aligning very similar images, where the brightness constancy assumption mostly holds. SIFT-Flow (Liu et al., 2010) was an early method that aimed at expanding optical flow-style approaches for matching pairs of images across physically distinct, and visually different scenes (and later generalized to joint image set alignment using cycle consistency (Zhou et al., 2015)). Some approaches such as SCV (Cech et al., 2010) and MODS (Mishkin et al., 2015), were proposed to grow matches around initial warping. In the deep era, (Long et al., 2014) showed that ConvNet activation features can be used for correspondence, achieving similar performance to SIFT-Flow. (Choy et al., 2016) proposed to learn matches with a Correspondence Contrastive loss, producing semi-dense matches. (Rocco et al., 2018b) introduced the idea of using 4D convolutions on the feature correlations to learn to filter neighbour consensus. Note that these latter works target semantic correspondences, whereas we focus on the case when all images depict the same physical scene.

Deep Flow methods. Deep networks can be trained to predict optical flow and to be robust to drastic appearance changes, but require adapted loss and architectures. Flows can be learned in a completely supervised way using synthetic data, e.g. in (Dosovitskiy et al., 2015; Ilg et al., 2017), but transfer to real data remains a difficult problem. Unsupervised training through reconstruction has been proposed in several works, targeting brightness consistency (Ahmadi and Patras, 2016; Wang et al., 2018a), gradient consistency (Ren et al., 2017) or high SSIM (Jason et al., 2016; Yin and Shi, 2018). This idea of learning correspondences through reconstruction has been applied to video, reconstructing colors (Vondrick et al., 2018), predicting weights

for frame reconstruction (Kong and Fowlkes, 2019; Lai and Xie, 2019), or directly optimizing feature consistency in the warped images (Wang et al., 2019). Several papers have introduced cycle consistency as an additional supervisory signal for image alignment (Zhou et al., 2016; Wang et al., 2019). Recently, feature correlation became a key part of several architectures (Ilg et al., 2017; Sun et al., 2018) aiming at predicting dense flows. Particularly relevant to us is the approach of (Melekhov et al., 2019) which includes a feature correlation layer in a U-Net (Ronneberger et al., 2015) architecture to improve flow resolution. A similar approach has been used in (Laskar et al., 2020) which predicts dense correspondences. Recently, Glu-Net (Prune et al., 2020) learns dense correspondences by investigating the combined use of global and local correlation layers.

Hybrid parametric/non-parametric image alignment. Classic "plane + parallax" approaches (Sawhney, 1994; Kumar et al., 1994; Irani et al., 2002; Wulff et al., 2017) aimed to combine parametric and non-parametric alignment by first estimating a homography (plane) and then considering the violations from that homography (parallax). Similar ideas also appeared in stereo, e.g. model-based stereo (Debevec et al., 1996). Recently, (Yin and Shi, 2018; Cao et al., 2019) proposed to learn optical flow by jointly optimizing with depth and ego-motion for stereo videos. Our RANSAC-Flow is also related to the methods designed for geometric multi-model fitting, such as RPA (Magri and Fusiello, 2014), T-linkage (Magri and Fusiello, 2017) and Progressive-X (Barath and Matas, 2019).

5.3 Method

Our two-stage RANSAC-Flow method is illustrated in Fig. 5.2. In this section, we describe the coarse alignment stage, the fine alignment stage, and how they can be iterated to use multiple homographies.

5.3.1 Coarse alignment by feature-based RANSAC

Our coarse parametric alignment is performed using RANSAC to fit a homography on a set of candidate sparse correspondences between the source and target images. We use off-the-shelf deep features (conv4 layer of a ResNet-50 network) to obtain these correspondences. We experimented with both pre-trained ImageNet features

as well as features learned via MoCo self-supervision (He et al., 2020), and obtained similar results. We found it was crucial to perform feature matching at different scales. We fixed the aspect ratio of each image and extracted features at seven scales: 0.5, 0.6, 0.88, 1, 1.33, 1.66 and 2. Matches that were not symmetrically consistent were discarded. The estimated homography is applied to the source image and the result is given together with the target image as input to our fine alignment. We report coarse-only baselines in Sec. 5.4 for both features as "*ImageNet (He et al., 2016)+H* " and "*MoCo (He et al., 2020)+H* ".

5.3.2 Fine alignment by local flow prediction

Given a source image I_s and a target image I_t which have already been coarsely aligned, we want to predict a fine flow $F_{s\to t}$ between them. We write $\mathbf{F}_{s\to t}$ as the mapping function associated to the flow $F_{s\to t}$. Since we only expect the fine alignment to work in image regions where the homography is a good approximation of the deformation, we predict a matchability mask $M_{s\to t}$, indicating which correspondences are valid. In the following, we first present our objective function, then how and why we optimize it using a self-supervised deep network.

Objective function. Our goal is to find a flow that warps the source into an image similar to the target. We formalize this by writing an objective function composed of three parts: a reconstruction loss $\mathcal{L}_{rec}$, a matchability loss $\mathcal{L}_m$ and a cycle-consistency loss $\mathcal{L}_c$. Given the pair of images(I_s, I_t) the total loss is:

$$\mathcal{L}(I_s, I_t) = \mathcal{L}_{rec}(I_s, I_t) + \lambda\mathcal{L}_m(I_s, I_t) + \mu\mathcal{L}_c(I_s, I_t) \tag{5.1}$$

with λ and μ hyper-parameters weighting the contribution of the matchability and cycle loss. We detail these three components in the following paragraphs. Each loss is defined pixel-wise.

Matchability loss. Our matchability mask can be seen as pixel-wise weights for the reconstruction and cycle-consistency losses. These losses will thus encourage the matchability to be zero. To counteract this effect, the matchability loss encourages the matchability mask to be close to one. Since the matchabiliy should be consistent between images, we define the cycle-consistent matchability at position (x,y) in I_t,

(x',y') in I_s with $(x, y) = \mathbf{F}_{s\to t}(x', y')$ as:

$$M_t^{cycle}(x, y) = M_{t\to s}(x, y)M_{s\to t}(x', y') \tag{5.2}$$

where $M_{s\to t}$ is the matchability predicted from source to target and $M_{t\to s}$ the one predicted from target to source. M_t^{cycle} will be high only if both the matchability of the corresponding pixels in the source and target are high. The matchability loss encourages this cycle-consistent matchability to be close to 1:

$$\mathcal{L}_m(I_s, I_t) = \sum_{(x,y)\in I_t} |M_t^{cycle}(x, y) - 1| \tag{5.3}$$

Note that directly encouraging the matchability to be 1 leads to similar quantitative results, but using the cycle consistent matchability helps to identify regions that are not matchable in the qualitative results.

Reconstruction loss. Reconstruction is the main term of our objective and is based on the idea that the source image warped with the predicted flow $F_{s\to t}$ should be aligned to the target image I_t. We use the structural similarity (SSIM) (Wang et al., 2004) as a robust similarity measure:

$$\mathcal{L}_{rec}^{SSIM}(I_s, I_t) = \sum_{(x,y)\in I_t} M_t^{cycle}(x, y)\,(1 - SSIM\,(I_s(x', y'), I_t(x, y))) \tag{5.4}$$

Cycle consistency loss. We enforce cycle consistency of the flow for 2-cycles:

$$\mathcal{L}_c(I_s, I_t) = \sum_{(x,y)\in I_t} M_t^{cycle}(x, y)\|(x', y'), \mathbf{F}_{t\to s}(x, y)\|_2 \tag{5.5}$$

Optimization with self-supervised network. Optimizing objective functions similar to the one described above is common to most optical flow approaches. However, this is known to be an extremely difficult task because of the highly non-convex nature of the objective which typically has many bad local minima. Recent works on the priors implicit within deep neural network architectures (Shocher et al., 2018; Ulyanov et al., 2018) suggest that optimizing the flow as the output of a neural network might overcome these problems. Unfortunately, our objective is still too complex to obtain good result from optimization on just a single image pair. We thus built a

larger database of image pairs on which we optimize the neural network parameters in a self-supervised way (i.e. without need for any annotations). The network could then be fine-tuned on the test image pair itself, but we have found that this single-pair optimization leads to unstable results. However, if several pairs similar to the test pair are available (i.e. we have access to the entire test set), the network can be fine-tuned on this test set which leads to some improvement, as can be seen in our experiments where we systematically report our results with and without fine-tuning.

To collect image pairs for the network training, we simply sample pairs of images representing the same scene and applied our coarse matching procedure. If it led to enough inliers, we added the pair to our training image set, if not we discarded it. For all the experiments, we sampled image pairs from the MegaDepth (Li and Snavely, 2018) scenes, using $20,000$ image pairs from 100 scenes for training and 500 pairs from 30 different scenes for validation.

5.3.3 Multiple homographies

The overall procedure described so far provides good results on image pairs where a single homography serves as a good (if not perfect) approximation of the overall transformation (e.g. planar scenes). This is, however, not the case for many image pairs with strong 3D effects or large objects displacements. To address this, we iterate our alignment algorithm to let it discover more homography candidates. At each iteration, we remove feature correspondences that were inliers for the previous homographies as well as from locations inside the previously predicted matchability masks, and recompute RANSAC again. We stop the procedure when not enough candidate correspondences remain. The full resulting flow is obtained by simply aggregating the estimated flows from each iteration together. The number of homographies considered depends on the input image pairs. For example, the average number of homographies we obtain from pairs for two-view geometry estimation in the YFCC100M (Thomee et al., 2016) dataset is about five. While more complex combinations could be considered, this simple approach provides surprisingly robust results. In our experiments, we quantitatively validate the benefits of using these multiple homographies ("*multi-H*").

5.3.4 Architecture and implementation details

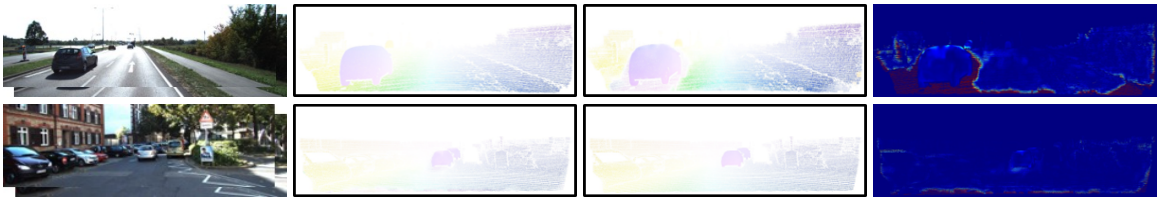
In our fine-alignment network, the input source and target images (I_s, I_t) are first processed separately by a fully-convolutional *feature extractor* which outputs two feature maps (f_s, f_t). Each feature from the source image is then compared to features in a $(2K+1) \times (2K+1)$ square neighbourhood in the target image using cosine similarity, similar to (Dosovitskiy et al., 2015; Ilg et al., 2017). This results in a $W \times H \times (2K+1)^2$ similarity tensor s defined by:

$$s(i, j, (m+K+1)(n+K+1)) = \frac{f_s(i,j).f_t(i-m, j-n)}{\|f_s(i,j)\|\|f_t(i-m, j-n)\|} \tag{5.6}$$

where $m, n \in [-K, ..., K]$ and "." denotes dot product. In all our experiments, we used $K = 3$. This similarity tensor is taken as input by two fully-convolutional *prediction networks* which predict flow and matchability.

Our *feature extractor* is similar to the *Conv3* feature extractor in ResNet-18 (He et al., 2016) but with minor modifications: the first 7×7 convolutional kernel of the network is replaced by a 3×3 kernel without stride and all the max-poolings and strided-convolution are replaced by their anti-aliasing versions proposed in (Zhang, 2019). These changes aim at reducing the loss of spatial resolution in the network, the output feature map being 1/8th of the resolution of the input images. The flow and matchability *prediction networks* are fully convolutional networks composed of three Conv+Relu+BN blocks (Convolution, Relu activation and Batch Normalization (Ioffe and Szegedy, 2015)) with 512, 256, 128 filters respectively and a final convolutional layer. The output flows and matchability are bilinearly upsampled to the resolution of the input images. Note we tried using up-convolutions, but this slightly decreased the performance while increasing the memory footprint.

We use Kornia (Riba et al., 2020) for homography warping. All images were resized so that their minimum dimension is 480 pixels. The hyper-parameters of our objective are set to $\lambda = 0.01$, $\mu = 1$. We provide a study of λ and μ in Sec. 5.4.4. The entire fine alignment model is learned from random initialization using the Adam optimizer (Kingma and Ba, 2015) with a learning rate of 2e-4 and momentum terms β_1, β_2 set to 0.5, 0.999. We trained only with $\mathcal{L}_{rec}$ for the first 150 epochs then added $\mathcal{L}_c$ for another 50 epochs and finally trained with all the losses (Eqn. 5.1) for the final 50 epochs. We use a mini-batch size of 16 for all the experiments. The whole training converged in approximately 30 hours using a single GPU Geforce GTX 1080 Ti for



(a) Input (b) Predicted (c) Ground truth (d) Error map

Figure 5.3: Visual results on KITTI (Menze and Geiger, 2015a). We show the inputs, predicted flow, ground-truth flow and the error map in (a), (b), (c) and (d) respectively.

Table 5.1: Dense correspondences evaluation on KITTI 2015 (Menze and Geiger, 2015a) and Hpatches (Balntas et al., 2017). We report the AEE (Average Endpoint Error) and Fl-all (Ratio of pixels where flow estimate is wrong by both 3 pixels and $\geq 5\%$). The computational time for EpicFlow and FlowField is 16s and 23s respectively, while our approach takes 4s.

Method	KITTI 2015 Train (AEE ↓) noc	KITTI 2015 Train (AEE ↓) all	KITTI 2015 Test (Fl-all ↓) noc	KITTI 2015 Test (Fl-all ↓) all	Hpatches Viewpoint (AEE ↓) 1	2	3	4	5
Supervised Approaches									
FlowNet2 (Ilg et al., 2017; Melekhov et al., 2019; Yin and Shi, 2018)	4.93	10.06	6.94	10.41	5.99	15.55	17.09	22.13	30.68
PWC-Net (Sun et al., 2018; Melekhov et al., 2019)	-	10.35	**6.12**	**9.60**	4.43	11.44	15.47	20.17	28.30
Rocco (Rocco et al., 2017; Melekhov et al., 2019)	-	-	-	-	9.59	18.55	21.15	27.83	35.19
DGC-Net (Melekhov et al., 2019)	-	-	-	-	1.55	5.53	8.98	11.66	16.70
DGC-Nc-Net (Laskar et al., 2020)	-	-	-	-	1.24	4.25	8.21	9.71	13.35
Glu-Net (Prune et al., 2020)	6.86	9.79	-	-	0.59	4.05	7.64	9.82	14.89
Weakly Supervised Approaches									
ImageNet (He et al., 2016) + H	13.49	17.26	-	-	1.33	3.34	3.71	6.04	10.07
Cao et al. (Cao et al., 2019)	4.19	5.13	-	-	-	-	-	-	-
Unsupervised Approaches									
Moco (He et al., 2020) + H	13.86	17.60	-	-	1.47	2.96	3.43	7.73	10.53
DeepMatching (Revaud et al., 2016; Melekhov et al., 2019)	-	-	-	-	5.84	4.63	12.43	12.17	22.55
DSTFlow (Ren et al., 2017)	6.96	16.79	-	39	-	-	-	-	-
GeoNet (Yin and Shi, 2018)	6.77	10.81	-	-	-	-	-	-	-
EpicFlow (Revaud et al., 2015; Yin and Shi, 2018)	4.45	**9.57**	16.69	26.29	-	-	-	-	-
FlowField (Bailer et al., 2015)	-	-	10.98	19.80	-	-	-	-	-
Moco Feature									
Ours	4.15	12.63	14.60	26.16	0.52	2.13	4.83	5.13	6.36
w/o fine-tuning	4.67	13.51	-	-	0.53	**2.04**	**2.32**	6.54	6.79
w/o Multi-H	7.04	14.02	-	-	-	-	-	-	-
ImageNet Feature									
Ours	**3.87**	12.48	14.12	25.76	**0.51**	2.36	2.91	**4.41**	**5.12**
w/o fine-tuning	4.55	13.51	-	-	**0.51**	2.37	2.64	4.49	5.16
w/o Multi-H	6.74	13.77	-	-	-	-	-	-	-

the 20k image pairs from the MegaDepth. For fine-tuning on the target dataset, we used a learning rate of 2e-4 for another 10K iterations.

5.4 Experiments

In this section, we evaluate our approach in terms of resulting correspondences (Sec. 5.4.1), downstream tasks (Sec 5.4.2), applications to texture transfer and artwork analysis (Sec. 5.4.3) as well as analysis of hyper-parameters in the objective function 5.1 (Sec. 5.4.4). We provide more visual results at `http://imagine.enpc.fr/~shenx/`

Table 5.2: Sparse correspondences evaluation on RobotCar (Maddern et al., 2017; Larsson et al., 2019) and MegaDepth (Li and Snavely, 2018). We report the accuracy over all annotated alignments for pixel error smaller than d pixels. All the images are resized to have minimum dimension 480 pixels.

Method	RobotCar Acc(≤ d pixels ↑)			MegaDepth Acc(≤ d pixels ↑)		
	1	3	5	1	3	5
ImageNet (He et al., 2016)+H	1.03	8.12	19.21	3.49	23.48	43.94
Moco (He et al., 2020)+H	1.08	8.77	20.05	3.70	25.12	45.45
SIFT-Flow (Liu et al., 2010)	1.12	8.13	16.45	8.70	12.19	13.30
NcNet (Rocco et al., 2018b)+H	0.81	7.13	16.93	1.98	14.47	32.80
DGC-Net (Melekhov et al., 2019)	1.19	9.35	20.17	3.55	20.33	34.28
Glu-Net (Prune et al., 2020)	**2.16**	**16.77**	**33.38**	25.2	51.0	56.8
Moco Feature						
Ours	2.10	16.07	31.66	**53.47**	83.45	**86.81**
w/o Multi-H	2.06	15.77	31.05	50.65	78.34	81.59
w/o Fine-tuning	2.09	15.94	31.61	52.60	**83.46**	86.80
ImageNet Feature						
Ours	2.10	16.09	31.80	53.15	83.34	86.74
w/o Multi-H	2.06	15.84	31.30	50.08	77.84	81.08
w/o Fine-tuning	2.09	16.00	31.90	52.80	83.31	86.64

RANSAC-Flow/.

5.4.1 Direct correspondences evaluation

Optical flow. We evaluate the quality of our dense flow on the KITTI 2015 flow (Menze and Geiger, 2015a) and Hpatches (Balntas et al., 2017) datasets and report the results in Table 5.1.

On KITTI (Menze and Geiger, 2015a), we evaluated both on the training and the test set since other approaches report results on one or the other. Note we could not perform an ablation study on the test set since the number of submissions to the online server is strictly limited. We report results both on non-occluded (noc) and all regions. Our results are on par with state of the art unsupervised and weakly supervised results on non-occluded regions, outperforming for example the recent approach (Cao et al., 2019; Prune et al., 2020). Unsurprisingly, our method is much weaker on occluded regions since our algorithm is not designed specifically for optical flow performances and has no reason to handle occluded regions in a good way. We find that the largest errors are actually in occluded regions and image boundaries (Fig. 5.3). Interestingly, our ablations show that the multiple homographies is critical

(a) Source (b) Target (c) Coarse align. (d) Fine align. (e) Flows

Figure 5.4: Visual results on RobotCar (Maddern et al., 2017) (1st row), Megadepth (Li and Snavely, 2018) (2nd row) and Hpatches (Balntas et al., 2017) (3rd row) using one homography. We show the source and target in (a), (b). The overlapped images after coarse and fine alignment are in (c) and (d) with zoomed details. The coarse (top) and fine (bottom) flows are in (e).

to our results even if the input images appear quite similar.

For completeness, we also present results on the Hpatches (Balntas et al., 2017). Note that Hpatches dataset is synthetically created by applying homographies to a set of real images, which would suggest that our coarse alignment alone should be enough. However, in practice, we have found that, due to the lack of feature correspondences, adding the fine flow network significantly boosts the results compared to using only our coarse approach. While these results show that our approach is reasonable, these datasets only contain very similar and almost aligned pairs while the main goal of our approach is to be able to handle challenging cases with strong viewpoint and appearance variations.

Sparse correspondences. Dense correspondence annotations are typically not available for extreme viewpoint and imaging condition variations. We thus evaluated our results on sparse correspondences available on the RobotCar (Maddern et al., 2017; Larsson et al., 2019) and MegaDepth (Li and Snavely, 2018) datasets. In Robotcar, we evaluated on the correspondences provided by (Larsson et al., 2019), which leads to approximately 340M correspondences. The task is especially challeng-

ing since the images correspond to different and challenging conditions (dawn, dusk, night, etc.) and most of the correspondences are on texture-less region such as roads where the reconstruction objective provides very little information. However, viewpoints in RobotCar are still very similar. To test our method on pairs of images with very different viewpoints, we used pairs of images from scenes of the MegaDepth (Li and Snavely, 2018) dataset that we didn't use for training and validation. Note that no real ground truth is available and we use as reference the result of SfM reconstructions. More precisely, we take 3D points as correspondences and randomly sample 1 600 pairs of images that shared more than 30 points, which results in approximately 367K correspondences.

On both datasets, we evaluated several baselines which provide dense correspondences and were designed to handle large viewpoint changes, inluding SIFT-Flow (Liu et al., 2010), variants of NcNet (Rocco et al., 2018b), DGC-Net (Melekhov et al., 2019) and the very recent, concurrently developed approach Glu-Net (Prune et al., 2020). In the results provided in Tab. 5.2, we can see that our approach is comparable to Glu-Net on RobotCar (Maddern et al., 2017; Larsson et al., 2019) but largely improves performances on MegeDepth (Li and Snavely, 2018). We believe this is because by the large viewpoint variations on MegeDepth is better handled by our method. This qualitative difference between the datasets can be seen in the visual results in Fig. 5.4. Note that we can clearly see the effect of fine flows on the zoomed details.

5.4.2 Evaluation for downstream tasks

Given the limitations of the correspondence benchmarks discussed in the previous paragraph, and to demonstrate the practical interest of our results, we now evaluate our correspondences on two standard geometry estimation benchmarks where many results from competing approaches exist. Note that competing approaches typically use only sparse matches for these tasks, and being able to perform them using dense correspondences is a demonstration of the strength and originality of our method.

Two-view geometry estimation. Given a pair of views of the same scene, two-view geometry estimation aims at recovering their relative pose. To validate our approach, we follow the standard setup of (Zhang et al., 2019) evaluating on 4×1000 image pairs for 4 scenes from YFCC100M (Thomee et al., 2016) dataset and reporting mAP for different thresholds on the angular differences between ground truth and

Table 5.3: Two-view geometric estimation on YFCC100M (Thomee et al., 2016; Zhang et al., 2019)

Method	mAP @5°	mAP@10°	mAP@20°
SIFT (Lowe, 2004)	46.83	68.03	80.58
Contextdesc (Luo et al., 2019)	47.68	69.55	**84.30**
Superpoint (DeTone et al., 2018)	30.50	50.83	67.85
PointCN (Moo Yi et al., 2018; Zhang et al., 2019)	47.98	-	-
PointNet++ (Qi et al., 2017b; Zhang et al., 2019)	46.23	-	-
N^3Net (Plötz and Roth, 2018; Zhang et al., 2019)	49.13	-	-
DFE (Ranftl and Koltun, 2018; Zhang et al., 2019)	49.45	-	-
OANet (Zhang et al., 2019)	52.18	-	-
Moco Feature			
Ours	**64.88**	**73.31**	81.56
w/o multi-H	61.10	70.50	79.24
w/o fine-tuning	63.48	72.93	81.59
ImageNet Feature			
Ous	62.45	70.84	78.99
w/o multi-H	59.90	68.80	77.31
w/o fine-tuning	62.10	70.78	79.07

predicted vectors for both rotation and translation as the error metric. For each image pair, we use the flow we predict in regions with high matchability (> 0.95) to estimate an essential matrix with RANSAC and the 5-point algorithm (Hartley and Zisserman, 2003). To avoid correspondences in the sky, we used the pre-trained the segmentation network provided in (Zhou et al., 2017) to remove them. While this require some supervision, this is reasonable since most of the baselines we compare to have been trained in a supervised way. As can be seen in Tab. 5.3, our method outperforms all the baselines by a large margin including the recent OANet (Zhang et al., 2019) method which is trained with ground truth calibration of cameras. Also note that using multiple homographies consistently boosts the performance of our method.

Once the relative pose of the cameras has been estimated, our correspondences can be used to perform stereo reconstruction from the image pair as illustrated in Fig. 5.1(c) and in the project webpage. Note that contrary to many stereo reconstruction methods, we can use two very different input images.

Day-Night Visual Localization. Another task we performed is visual localization. We evaluate on the local feature challenge of the Visual Localization benchmark (Sattler et al., 2018, 2012). For each of the 98 night-time images contained in

Table 5.4: Visual Localization on Aachen night-time (Sattler et al., 2018, 2012).

Method	0.5m,2°	1m,5°	5m,10°
Upright RootSIFT (Lowe, 2004)	36.7	54.1	72.5
DenseSfM (Sattler et al., 2018)	39.8	60.2	84.7
HAN + HN++ (Mishchuk et al., 2017; Mishkin et al., 2018)	39.8	61.2	77.6
Superpoint (DeTone et al., 2018)	42.8	57.1	75.5
DELF (Noh et al., 2017)	39.8	61.2	85.7
D2-net (Dusmanu et al., 2019)	44.9	66.3	**88.8**
R2D2 (Revaud et al., 2019b)	**45.9**	66.3	**88.8**
Moco Feature			
Ours	44.9	**68.4**	**88.8**
w/o Multi-H	42.9	**68.4**	**88.8**
w/o Fine-tuning	41.8	**68.4**	**88.8**
ImageNet Feature			
Ous	44.9	**68.4**	**88.8**
w/o Multi-H	43.9	66.3	**88.8**
w/o Fine-tuning	44.9	**68.4**	**88.8**

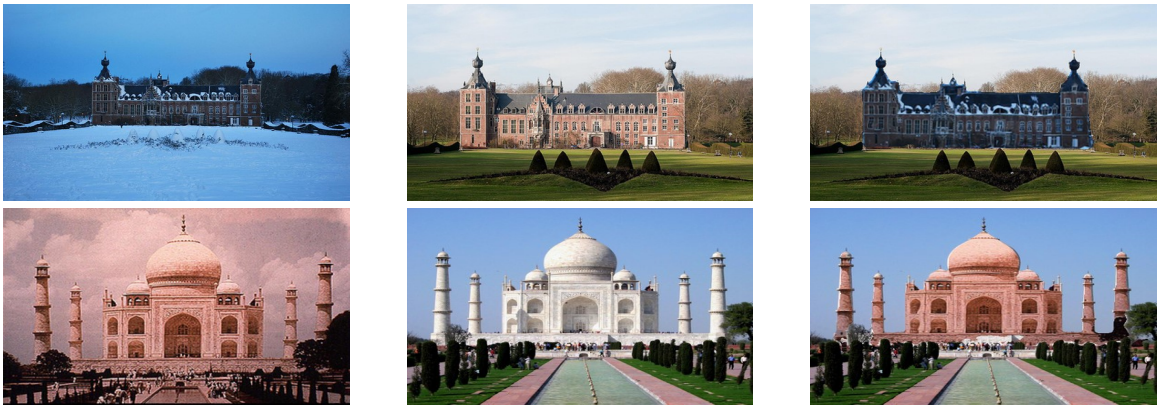
the dataset, up to 20 relevant day-time images with known camera poses are given. We followed evaluation protocol from (Sattler et al., 2018) and first compute image matching for a list of image pairs and then give them as input to COLMAP (Schonberger and Frahm, 2016) that provides a localisation estimation for the queries. To limit the number of correspondences we use only correspondences on a sparse set of keypoints using the Superpoint (DeTone et al., 2018). Our results are reported in Tab. 5.4 and are on par with state of the art results.

5.4.3 Applications

One of the most exciting aspect of our approach is that it enables new applications based on the fine alignment of historical, internet or artistic images.

Texture transfer. Our approach can be used to transfer texture between images. In Figure 5.5 and 5.1(f) we show results using historical and modern images from the LTLL dataset (Fernando et al., 2015). We use the pre-trained segmentation network of (Zhou et al., 2018), and transfer the texture from the source to the target building regions.

Internet images alignment. As visualized in Fig. 5.1(d) and Fig. 5.6, we can align sets of internet images, similar to (Shrivastava et al., 2011). Even if our image



(a) Source (b) Target (c) Texture transfer

Figure 5.5: Texture transfer : (a) source, (b) target and (c) texture transferred result.

(a) Source (b) Coarse align. (c) Fine align.

Figure 5.6: Aligning a group of Internet images from the Medici Fountain, similar to (Shrivastava et al., 2011). We show the source images (a), the average image after coarse (b) and fine alignment (c). The GIF animation can be found in our project webpage http://imagine.enpc.fr/~shenx/RANSAC-Flow/.

set is not precisely the same, much more details can be seen in the average of our fine-aligned images.

Artwork analysis. Finding and matching near-duplicate patterns is an important problem for art historians. Computationally, it is difficult because the duplicate appearance can be very different (Shen et al., 2019). In Fig. 5.7, we show visual results of aligning different versions of artworks from the Brueghel dataset (Shen et al., 2019) with our coarse and fine alignment. We can clearly see that a simple homography is not sufficient and that the fine alignment improves results by identifying complex displacements. The fine flow can thus be used to provide insights on Brueghel's copy process. Indeed, we found that some artworks were copied in a spatially consistent way, while in others, different parts of the picture were not aligned with each other. This can be clearly seen in the flows in Fig. 5.9, which are either very regular or

(a) Inputs (b) W/o align. (c) Coarse align. (d) Fine align. (e) Flows

Figure 5.7: Aligning pairs of similar artworks from the Brueghel (bru): We show the pairs in (a). The average images without alignment, after coarse and fine alignment are in (b), (c) and (d). The coarse (top) and fine (bottom) flows are in (e).

very discontinuous. The same process can be applied to more than a single pair of images, as illustrated in Fig. 5.1(e) and 5.8 where we align together many similar details identified by (Shen et al., 2019). Visualizing the succession of the finely aligned images allows to identify their differences.

5.4.4 Dependency on λ and μ

Our training has 3 stages (Sec. 5.3.4): the model was firstly learned with the reconstruction loss $\mathcal{L}_{rec}$ then added cycle-consistent flow loss $\mathcal{L}_c$ and finally trained with all the losses (Eqn. 5.1). In Tab. 5.5, we provide an analysis on the weighting parameters λ and μ on sparse correspondences evaluation on MegaDepth (Li and Snavely, 2018) and report the accuracy at 3 pixels. We can see the stage 2 is not very sensitive with respect to μ (Tab. 5.5a), while the stage 3 with adding the mask loss is slightly more sensitive (Tab. 5.5b). Note that we then use the same parameters for fine-tuning on

(a) Source (b) (Shen et al., 2019) (c) Coarse align. (d) Fine align.

Figure 5.8: Aligning details discovered by (Shen et al., 2019): (a) sources; average from (Shen et al., 2019) (b), with coarse (c) and fine (d) alignment; The GIF animation can be found in our project webpage http://imagine.enpc.fr/~shenx/RANSAC-Flow/.

Table 5.5: Dependency on λ and μ, we evaluate on sparse correspondences on MegaDepth (Li and Snavely, 2018) and report the accuracy at 3 pixels. (a) Training stage 2: dependency on μ with $\lambda = 0$; (b) Training stage 3: dependency on λ with $\mu = 1$ (optimal in Tab. 5.5a).

μ	Acc. ($\leq$ 3 pixels)
2	78.2
1	**78.3**
0.5	**78.3**

(a) Training stage 2: dependency on μ with $\lambda = 0$.

λ	Acc. ($\leq$ 3 pixels)
0.02	83.0
0.01	**83.5**
0.005	80.5

(b) Training stage 3: dependency on λ with $\mu = 1$ (optimal in Tab. 5.5a).

the different datasets.

Figure 5.9: Analyzing copy process from flow. The flow is smooth from the middle to the right one, while it is irregular from the middle to the left one.

5.5 Conclusion

We have introduced a new unsupervised method for generic dense image alignment which performs well on a wide range of tasks. Our main insight is to combine the advantages of parametric and non-parametric methods in a two-stage approach and to use multiple homography estimations as initializations for fine flow prediction. We also demonstrated it allows new applications for artwork analysis.

Chapter 6

Conclusion

In this chapter, we summarize the main contributions of this thesis and outline some research directions for future work.

6.1 Summary of Contributions

In this thesis, we focused on three tasks for artwork analysis: style-invariant patterns retrieval, near-duplicated patterns discovery, and dense image alignment. We have investigated three methods to solve these problems. The technical contributions are detailed below.

In Chapter 3, we proposed a self-supervised feature fine-tuning strategy to address style-invariant feature matching. Our approach consists of leveraging spatial consistency to identify positive and negative training pairs. We optimized a standard triplet loss on these training samples. Our approach can learn features adapted to the training set for matching. In terms of experimental results, we showed that our approach can match features in artworks across different styles. Our approach also improved performance on one-shot art detail detection on (bru; Shen et al., 2019) as well as geo-localization datasets: LTLL (Fernando et al., 2015) and Oxford (Philbin et al., 2007).

In Chapter 4, we proposed an approach of learning co-segmentation from synthetic data. To generate the training data, we blended objects to background images using style transfer (Huang and Belongie, 2017) and Poisson blending Pérez et al. (2003). We also designed an architecture based on transformer (Vaswani et al., 2017) for co-segmentation. The network was learned using masks and correspondences of

the blended objects. We also validated the choices of our objective functions and the process of the data generation using Sparse Nc-Net (Rocco et al., 2020). Our approach has been evaluated on various tasks including one-shot art detail detection on Brueghel (Shen et al., 2019; bru) and place recognition on the Pitts30k (Torii et al., 2013) and Tokyo 24/7 (Torii et al., 2015) datasets.

In Chapter 5, we proposed an unsupervised approach for robust dense image alignment. Our approach is a two-stage approach:: in the first stage, we extract multi-scale features and match them to estimate Homography transformations using RANSAC (Fischler and Bolles, 1981); in the second stage, we designed and learned a Convolutional Neural Network (CNN) through the optimization of a reconstruction loss between target images and warped source images. Our approach is without any supervision. We showed that our approach performs well on various tasks, including standard optical flow estimation on KITTI (Menze and Geiger, 2015a), sparse correspondences evaluation on RobotCar (Maddern et al., 2017; Larsson et al., 2019) and MegaDepth (Li and Snavely, 2018), two-view geometry estimation on YFCC (Thomee et al., 2016), and 3D reconstruction on Aachen night-time (Sattler et al., 2018, 2012)

6.2 Future Work

In this section, we analyze some future research directions that could extend the works presented in this thesis.

Interactive annotation system incorporating unsupervised / weakly supervised techniques. As presented in Chapter. 1, the main challenge of artwork analysis is the lack of training data and the main application for object discovery is to automatically collect datasets, it would be useful and efficient to develop annotations tools that can use unsupervised or weakly supervised algorithms. Note that many annotation tools have been developed and made a significant impact on scientific research, representative works are VGG Image Annotator (VIA) (Dutta et al., 2016) and LabelMe (Russell et al., 2008). However, these tools only provide a user interface for annotators and no machine learning algorithms are involved. Another advantage is that the task will be better defined by users. In this thesis, we focus on retrieving or detecting near-duplicated patterns. However, for some art historians, obtaining semantically similar patterns might be also interesting. With the system, the task will be better defined and fit to personal interests.

This direction seems to be too engineering as a research topic, but there indeed exists some practical problems that are interesting for scientific research. For example, if annotators have a fixed budget, such as labeling 10 samples per class, how to leverage algorithms to recommend these samples to annotators. If annotators label bounding boxes, how to find a robust and efficient way to obtain pixel-level annotations. Besides, as new annotations come, how to efficiently update a model without catastrophic forgetting (Kirkpatrick et al., 2017; Kemker et al., 2018) etc.

Learning from synthetic data and weakly supervised pairs. In Chapter 4, we showed that it is possible to learn co-segmentation from synthetic data. Moreover, the learned models achieved promising results on different tasks. One interesting direction would be how to leverage weakly supervised pairs and synthetic data during the training. Positive pairs, which contain repeated segments, are not complicated to obtain. One solution has been presented in Chapter 5, which consists of leveraging multi-scale feature matching and geometric verification. Note that this weakly supervised learning can be done on test sets. Then the problem is highly related to the topic of test time fine-tuning (Sun et al., 2020).

End-to-end dense image alignment. In Chapter 5, we presented a two-stage method for unsupervised dense image alignment. The first stage consists of using RANSAC to estimate Homography transformations, which is not differentiable. As future work, it would be interesting to adapt this coarse-to-fine regime such that the overall approach is fully differentiable. As predicting matchable regions significantly improves the accuracy of the correspondence (in Sec. 5.4.4), It might be interesting to learn both matchable regions and optical flow end-to-end, which is different from other works on learning optical flow. One possible solution would be designing a proper synthetic dataset for training.

Bibliography

Brueghel family: Jan brueghel the elder." the brueghel family database. university of california, berkeley. http://www.janbrueghel.net/. Accessed: 2018-10-16.

Docexplore. https://www.docexplore.eu/.

Imgsai. https://imgs.ai/.

Timemachine. diamond.timemachine.eu.

Wikiart. https://www.wikiart.org/, a. Accessed: 2018-10-16.

Wikiart retriever. https://github.com/lucasdavid/wikiart/, b. Accessed: 2018-10-16.

A. Ahmadi and I. Patras. Unsupervised convolutional neural networks for motion estimation. In *International Conference on Image Processing (ICIP)*, 2016.

R. Arandjelović and A. Zisserman. Three things everyone should know to improve object retrieval. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

R. Arandjelović and A. Zisserman. Dislocation: Scalable descriptor distinctiveness for location recognition. In *Asian Conference on Computer Vision (ACCV)*, 2014.

R. Arandjelovic, P. Gronat, A. Torii, T. Pajdla, and J. Sivic. Netvlad: Cnn architecture for weakly supervised place recognition. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

M. Aubry, B. C. Russell, and J. Sivic. Painting-to-3d model alignment via discriminative visual elements. *ACM Transactions on Graphics (ToG)*, 2014.

A. Babenko, A. Slesarev, A. Chigorin, and V. Lempitsky. Neural codes for image retrieval. In *European Conference on Computer Vision (ECCV)*, 2014.

C. Bailer, B. Taetz, and D. Stricker. Flow fields: Dense correspondence fields for highly accurate large displacement optical flow estimation. In *International Conference on Computer Vision (ICCV)*, 2015.

V. Balntas, K. Lenc, A. Vedaldi, and K. Mikolajczyk. Hpatches: A benchmark and evaluation of handcrafted and learned local descriptors. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

D. Barath and J. Matas. Graph-cut ransac. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

D. Barath and J. Matas. Progressive-x: Efficient, anytime, multi-model fitting algorithm. In *International Conference on Computer Vision (ICCV)*, 2019.

D. Barath, J. Matas, and J. Noskova. Magsac: marginalizing sample consensus. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

O. Bounou, T. Monnier, I. Pastrolin, X. Shen, C. Benevent, M.-F. Limon-Bonnet, F. Bougard, M. Aubry, M. H. Smith, O. Poncet, et al. A web application for watermark recognition. *Journal of Data Mining and Digital Humanities*, 2020.

T. Brox, C. Bregler, and J. Malik. Large displacement optical flow. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.

Z. Cao, A. Kar, C. Hane, and J. Malik. Learning independent object motion from unlabelled stereoscopic videos. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko. End-to-end object detection with transformers. In *European Conference on Computer Vision (ECCV)*, 2020.

M. Caron, H. Touvron, I. Misra, H. Jégou, J. Mairal, P. Bojanowski, and A. Joulin. Emerging properties in self-supervised vision transformers. In *nternational Conference on Computer Vision (ICCV)*, 2021.

G. Castellano and G. Vessio. Towards a tool for visual link retrieval and knowledge discovery in painting datasets. In *Italian Research Conference on Digital Libraries*, 2020.

G. Castellano and G. Vessio. Deep convolutional embedding for digitized painting clustering. In *International Conference on Pattern Recognition (ICPR)*, 2021.

G. Castellano, E. Lella, and G. Vessio. Visual link retrieval and knowledge discovery in painting datasets. *Multimedia Tools and Applications*, 2021.

J. Cech, J. Matas, and M. Perdoch. Efficient sequential correspondence selection by cosegmentation. *Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2010.

H.-S. Chang and Y.-C. F. Wang. Optimizing the decomposition for multiple foreground cosegmentation. *Computer Vision and Image Understanding (CVIU)*, 2015.

L. Chen and J. Yang. Recognizing the style of visual arts via adaptive cross-layer correlation. In *ACM Multimedia*, 2019.

X. Chen, A. Shrivastava, and A. Gupta. Enriching visual knowledge bases via object discovery and segmentation. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

X. Chen, H. Fan, R. Girshick, and K. He. Improved baselines with momentum contrastive learning. *arXiv*, 2020a.

Y.-C. Chen, Y.-Y. Lin, M.-H. Yang, and J.-B. Huang. Show, match and segment: Joint weakly supervised learning of semantic matching and object co-segmentation. *Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2020b.

M.-M. Cheng, N. J. Mitra, X. Huang, P. H. Torr, and S.-M. Hu. Global contrast based salient region detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI*, 2014.

M. Cho, S. Kwak, C. Schmid, and J. Ponce. Unsupervised object discovery and localization in the wild: Part-based matching with bottom-up region proposals. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

C. B. Choy, J. Gwak, S. Savarese, and M. Chandraker. Universal correspondence network. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2016.

E. J. Crowley and A. Zisserman. Of gods and goats: Weakly supervised learning of figurative art. In *British Machine Vision Conference (BMVC)*, 2013.

E. J. Crowley and A. Zisserman. In search of art. In *European Conference on Computer Vision (ECCV)*, 2014.

E. J. Crowley and A. Zisserman. The art of detection. In *European Conference on Computer Vision (ECCV)*, 2016.

E. J. Crowley, O. M. Parkhi, and A. Zisserman. Face painting: querying art with photos. In *British Machine Vision Conference (BMVC)*, 2015.

N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2005.

A. David. From brueghel to warhol: Ai enters the attribution fray. *Nature*, 2019.

P. E. Debevec, C. J. Taylor, and J. Malik. Modeling and rendering architecture from photographs: A hybrid geometry-and image-based approach. In *The Proceedings of the ACM in Computer Graphics and Interactive Techniques (PACMCGIT)*, 1996.

J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.

D. DeTone, T. Malisiewicz, and A. Rabinovich. Superpoint: Self-supervised interest point detection and description. In *Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2018.

C. Doersch, S. Singh, A. Gupta, J. Sivic, and A. Efros. What makes paris look like paris? *ACM Transactions on Graphics (ToG)*, 2012.

C. Doersch, A. Gupta, and A. A. Efros. Mid-level visual element discovery as discriminative mode seeking. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2013.

C. Doersch, A. Gupta, and A. A. Efros. Context as supervisory signal: Discovering objects with predictable context. In *European Conference on Computer Vision (ECCV)*, 2014.

C. Doersch, A. Gupta, and A. A. Efros. Unsupervised visual representation learning by context prediction. In *International Conference on Computer Vision (ICCV)*, 2015.

A. Dosovitskiy, P. Fischer, E. Ilg, P. Hausser, C. Hazirbas, V. Golkov, P. Van Der Smagt, D. Cremers, and T. Brox. Flownet: Learning optical flow with convolutional networks. In *International Conference on Computer Vision (ICCV)*, 2015.

V. Dovgalecs, A. Burnett, P. Tranouez, S. Nicolas, and L. Heutte. Spot it! finding words and patterns in historical documents. In *International Conference on Document Analysis and Recognition (ICDAR)*, 2013.

M. Dusmanu, I. Rocco, T. Pajdla, M. Pollefeys, J. Sivic, A. Torii, and T. Sattler. D2-net: A trainable cnn for joint description and detection of local features. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

A. Dutta, A. Gupta, and A. Zissermann. VGG image annotator (VIA). http://www.robots.ox.ac.uk/~vgg/software/via/, 2016.

A. Elgammal, B. Liu, M. Elhoseiny, and M. Mazzone. Can: Creative adversarial networks, generating" art" by learning about styles and deviating from style norms. *International Conference on Computational Creativity (ICCC)*, 2017.

S. En, S. Nicolas, C. Petitjean, F. Jurie, and L. Heutte. New public dataset for spotting patterns in medieval document images. *Journal of Electronic Imaging*, 2016.

M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision (IJCV)*, 2010.

B. Fernando, T. Tommasi, and T. Tuytelaars. Location recognition over large time lags. *Computer Vision and Image Understanding (CVIU)*, 2015.

M. A. Fischler and R. C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 1981.

L. Gatys, A. S. Ecker, and M. Bethge. Texture synthesis using convolutional neural networks. 2015.

L. A. Gatys, A. S. Ecker, and M. Bethge. Image style transfer using convolutional neural networks. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

Y. Ge, H. Wang, F. Zhu, R. Zhao, and H. Li. Self-supervising fine-grained region similarities for large-scale image localization. In *European Conference on Computer Vision (ECCV)*, 2020.

A. Geiger, P. Lenz, and R. Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

R. Geirhos, P. Rubisch, C. Michaelis, M. Bethge, F. A. Wichmann, and W. Brendel. Imagenet-trained cnns are biased towards texture; increasing shape bias improves accuracy and robustness. In *International Conference on Learning Representations (ICLR)*, 2019.

G. Ghiasi, Y. Cui, A. Srinivas, R. Qian, T.-Y. Lin, E. D. Cubuk, Q. V. Le, and B. Zoph. Simple copy-paste is a strong data augmentation method for instance segmentation. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

S. Ginosar, D. Haas, T. Brown, and J. Malik. Detecting people in cubist art. In *European Conference on Computer Vision Worshops (ECCVW)*, 2014.

S. Ginosar, X. Shen, K. Dwivedi, E. Honig, and M. Aubry. The burgeoning computer-art symbiosis. *XRDS: Crossroads, The ACM Magazine for Students*, 2018.

R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

N. Gonthier, Y. Gousseau, S. Ladjal, and O. Bonfait. Weakly supervised object detection in artworks. In *European Conference on Computer Vision Worshops (ECCVW)*, 2018.

I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2014.

A. Gordo, J. Almazan, J. Revaud, and D. Larlus. End-to-end learning of deep visual representations for image retrieval. *International Journal of Computer Vision (IJCV)*, 2017.

P. Gronat, G. Obozinski, J. Sivic, and T. Pajdla. Learning and calibrating per-location classifiers for visual place recognition. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013.

E. Gultepe, T. E. Conturo, and M. Makrehchi. Predicting and grouping digitized paintings by style using unsupervised feature learning. *Journal of cultural heritage*, 2018.

R. Hartley and A. Zisserman. *Multiple view geometry in computer vision*. Cambridge university press, 2003.

A. Hati, S. Chaudhuri, and R. Velmurugan. Image co-segmentation using maximum common subgraph matching and region co-growing. In *European Conference on Computer Vision (ECCV)*, 2016.

D. C. Hauagge and N. Snavely. Image matching using local symmetry features. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

S. Hausler, S. Garg, M. Xu, M. Milford, and T. Fischer. Patch-netvlad: Multi-scale fusion of locally-global descriptors for place recognition. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick. Momentum contrast for unsupervised visual representation learning. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

A. Hertzmann. Can computers create art? In *Arts*. Multidisciplinary Digital Publishing Institute, 2018.

A. Hertzmann, C. E. Jacobs, N. Oliver, B. Curless, and D. H. Salesin. Image analogies. In *The Proceedings of the ACM in Computer Graphics and Interactive Techniques (PACMCGIT)*, 2001.

D. L. Hill, P. G. Batchelor, M. Holden, and D. J. Hawkes. Medical image registration. *Physics in medicine & biology*, 2001.

E. Honig. *Jan Brueghel and the Senses of Scale*. Pennsylvania State University Press, 2016.

K.-J. Hsu, Y.-Y. Lin, Y.-Y. Chuang, et al. Co-attention cnns for unsupervised object co-segmentation. In *International Joint Conference on Artificial Intelligence (IJCAI)*, 2018.

S. X. Hu, P. G. Moreno, Y. Xiao, X. Shen, G. Obozinski, N. D. Lawrence, and A. Damianou. Empirical bayes transductive meta-learning with synthetic gradients. In *International Conference on Learning Representations (ICLR)*, 2020.

Y. Hu, R. Song, and Y. Li. Efficient coarse-to-fine patchmatch for large displacement optical flow. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

X. Huang and S. Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. In *International Conference on Computer Vision (ICCV)*, 2017.

E. Ilg, N. Mayer, T. Saikia, M. Keuper, A. Dosovitskiy, and T. Brox. Flownet 2.0: Evolution of optical flow estimation with deep networks. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International Conference on Machine Learning (ICML)*, 2015.

M. Irani, P. Anandan, and M. Cohen. Direct recovery of planar-parallax from multiple frames. *Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2002.

M. Jaderberg, K. Simonyan, A. Zisserman, et al. Spatial transformer networks. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2015.

Y. Jae Lee, A. A. Efros, and M. Hebert. Style-aware mid-level representation for discovering visual connections in space and time. In *International Conference on Computer Vision (ICCV)*, 2013.

J. Janai, F. Guney, A. Ranjan, M. Black, and A. Geiger. Unsupervised learning of multi-frame optical flow with occlusions. In *European Conference on Computer Vision (ECCV)*, 2018.

J. Y. Jason, A. W. Harley, and K. G. Derpanis. Back to basics: Unsupervised learning of optical flow via brightness constancy and motion smoothness. In *European Conference on Computer Vision (ECCV)*, 2016.

H. Jégou, M. Douze, C. Schmid, and P. Pérez. Aggregating local descriptors into a compact image representation. In *Conference on Computer Vision and Pattern Recognition*, 2010.

K. R. Jerripothula, J. Cai, and J. Yuan. Image co-segmentation via saliency co-fusion. *Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, 2016.

K. R. Jerripothula, J. Cai, J. Lu, and J. Yuan. Object co-skeletonization with co-segmentation. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

W. Jiang, E. Trulls, J. Hosang, A. Tagliasacchi, and K. M. Yi. Cotr: Correspondence transformer for matching across images. In *International Conference on Computer Vision (ICCV)*, 2021.

J. Johnson, A. Alahi, and L. Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *European Conference on Computer Vision (ECCV)*, 2016.

A. Joulin, F. Bach, and J. Ponce. Discriminative clustering for image co-segmentation. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010.

A. Joulin, F. Bach, and J. Ponce. Multi-class cosegmentation. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

R. Kaoua, X. Shen, A. Durr, S. Lazaris, D. Picard, and M. Aubry. Image collation: Matching illustrations in manuscripts. In *International Conference on Document Analysis and Recognition (ICDAR)*, 2021.

S. Karayev, M. Trentacoste, H. Han, A. Agarwala, T. Darrell, A. Hertzmann, and H. Winnemoeller. Recognizing image style. *British Machine Vision Conference (BMVC)*, 2014.

R. Kemker, M. McClure, A. Abitino, T. Hayes, and C. Kanan. Measuring catastrophic forgetting in neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2018.

G. Kim, E. P. Xing, L. Fei-Fei, and T. Kanade. Distributed cosegmentation via submodular optimization on anisotropic diffusion. In *International Conference on Computer Vision (ICCV)*, 2011.

H. J. Kim, E. Dunn, and J.-M. Frahm. Learned contextual feature reweighting for image geo-localization. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017a.

S. Kim, D. Min, B. Ham, S. Jeon, S. Lin, and K. Sohn. Fcss: Fully convolutional self-similarity for dense semantic correspondence. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017b.

S. Kim, S. Lin, S. R. JEON, D. Min, and K. Sohn. Recurrent transformer networks for semantic correspondence. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2018.

S. Kim, D. Min, S. Jeong, S. Kim, S. Jeon, and K. Sohn. Semantic attribute matching networks. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*, 2015.

J. Kirkpatrick, R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A. A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska, et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences*, 2017.

S. Kong and C. Fowlkes. Multigrid predictive filter flow for unsupervised learning on videos. *arXiv*, 2019.

A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2012.

R. Kumar, P. Anandan, and K. Hanna. Direct recovery of shape from multiple views: A parallax based approach. In *International Conference on Pattern Recognition (ICPR)*, 1994.

Z. Lai and W. Xie. Self-supervised learning for video correspondence flow. In *British Machine Vision Conference (BMVC)*, 2019.

M. Larsson, E. Stenborg, L. Hammarstrand, M. Pollefeys, T. Sattler, and F. Kahl. A cross-season correspondence dataset for robust semantic segmentation. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

Z. Laskar, I. Melekhov, H. R. Tavakoli, J. Ylioinas, and J. Kannala. Geometric image correspondence verification by dense pixel matching. In *Winter Conference on Applications of Computer Vision (WACV)*, 2020.

C. Lee, W.-D. Jang, J.-Y. Sim, and C.-S. Kim. Multiple random walkers and their application to image cosegmentation. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

M. Leordeanu and M. Hebert. A spectral technique for correspondence problems using pairwise constraints. In *International Conference on Computer Vision (ICCV)*, 2005.

B. Li, Z. Sun, Q. Li, Y. Wu, and A. Hu. Group-wise deep object co-segmentation with co-attention recurrent neural network. In *International Conference on Computer Vision (ICCV)*, 2019.

D. Li, Y. Yang, Y.-Z. Song, and T. M. Hospedales. Deeper, broader and artier domain generalization. In *International Conference on Computer Vision (ICCV)*, 2017.

W. Li, O. H. Jafari, and C. Rother. Deep object co-segmentation. In *Asian Conference on Computer Vision (ACCV)*, 2018.

Z. Li and N. Snavely. Megadepth: Learning single-view depth prediction from internet photos. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. In *European Conference on Computer Vision (ECCV)*, 2014.

C. Liu, J. Yuen, and A. Torralba. Sift flow: Dense correspondence across scenes and its applications. *Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2010.

L. Liu, H. Li, and Y. Dai. Stochastic attraction-repulsion embedding for large scale image localization. In *International Conference on Computer Vision (ICCV)*, 2019.

J. L. Long, N. Zhang, and T. Darrell. Do convnets learn correspondence? In *Conference on Neural Information Processing Systems (NeurIPS)*, 2014.

D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision (IJCV)*, 2004.

B. D. Lucas, T. Kanade, et al. An iterative image registration technique with an application to stereo vision. In *International Joint Conference on Artificial Intelligence (IJCAI)*, 1981.

Z. Luo, T. Shen, L. Zhou, S. Zhu, R. Zhang, Y. Yao, T. Fang, and L. Quan. Geodesc: Learning local descriptors by integrating geometry constraints. In *European Conference on Computer Vision (ECCV*, 2018.

Z. Luo, T. Shen, L. Zhou, J. Zhang, Y. Yao, S. Li, T. Fang, and L. Quan. Contextdesc: Local descriptor augmentation with cross-modality context. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

W. Maddern, G. Pascoe, C. Linegar, and P. Newman. 1 year, 1000 km: The oxford robotcar dataset. *The International Journal of Robotics Research*, 2017.

L. Magri and A. Fusiello. T-linkage: A continuous relaxation of j-linkage for multi-model fitting. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

L. Magri and A. Fusiello. Multiple structure recovery via robust preference analysis. *Image and Vision Computing*, 2017.

H. Mao, M. Cheung, and J. She. Deepart: Learning joint representations of visual arts. In *ACM Multimedia*, 2017.

I. Melekhov, A. Tiulpin, T. Sattler, M. Pollefeys, E. Rahtu, and J. Kannala. Dgc-net: Dense geometric correspondence network. In *Winter Conference on Applications of Computer Vision (WACV)*, 2019.

T. Mensink and J. Van Gemert. The rijksmuseum challenge: Museum-centered visual recognition. In *Proceedings of International Conference on Multimedia Retrieval (ICMR)*. ACM, 2014.

M. Menze and A. Geiger. Object scene flow for autonomous vehicles. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015a.

M. Menze and A. Geiger. Object scene flow for autonomous vehicles. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015b.

A. Mishchuk, D. Mishkin, F. Radenovic, and J. Matas. Working hard to know your neighbor's margins: Local descriptor learning loss. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2017.

D. Mishkin, J. Matas, and M. Perdoch. Mods: Fast and robust method for two-view matching. *Computer Vision and Image Understanding (CVIU)*, 2015.

D. Mishkin, F. Radenovic, and J. Matas. Repeatability is not enough: Learning affine regions via discriminability. In *European Conference on Computer Vision (ECCV)*, 2018.

K. Moo Yi, E. Trulls, Y. Ono, V. Lepetit, M. Salzmann, and P. Fua. Learning to find good correspondences. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

A. Ng, M. Jordan, and Y. Weiss. On spectral clustering: Analysis and an algorithm. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2001.

D. Nister and H. Stewenius. Scalable recognition with a vocabulary tree. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2006.

H. Noh, A. Araujo, J. Sim, T. Weyand, and B. Han. Large-scale image retrieval with attentive deep local features. In *International Conference on Computer Vision (ICCV)*, 2017.

D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, and A. A. Efros. Context encoders: Feature learning by inpainting. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

P. Pérez, M. Gangnet, and A. Blake. Poisson image editing. In *Conference on Computer Graphics and Interactive Techniques (SIGGRAPH)*, 2003.

F. Perronnin and C. Dance. Fisher kernels on visual vocabularies for image categorization. In *Conference on Computer Vision and Pattern Recognition (2007)*, 2007.

J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman. Object retrieval with large vocabularies and fast spatial matching. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2007.

D. Picard, P.-H. Gosselin, and M.-C. Gaspard. Challenges in content-based image indexing of cultural heritage collections. *IEEE Signal Processing Magazine*, 2015.

T. Plötz and S. Roth. Neural nearest neighbors networks. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2018.

T. Prune, D. Martin, and T. Radu. GLU-Net: Global-local universal network for dense flow and correspondences. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

C. R. Qi, H. Su, K. Mo, and L. J. Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017a.

C. R. Qi, L. Yi, H. Su, and L. J. Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2017b.

R. Quan, J. Han, D. Zhang, and F. Nie. Object co-segmentation via graph optimized-flexible manifold ranking. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

F. Radenović, G. Tolias, and O. Chum. Fine-tuning cnn image retrieval with no human annotation. *Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2018.

R. Raguram, O. Chum, M. Pollefeys, J. Matas, and J.-M. Frahm. Usac: a universal framework for random sample consensus. *Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2012.

R. Ranftl and V. Koltun. Deep fundamental matrix estimation. In *European Conference on Computer Vision (ECCV)*, 2018.

S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2015.

Z. Ren, J. Yan, B. Ni, B. Liu, X. Yang, and H. Zha. Unsupervised deep learning for optical flow estimation. In *Association for the Advancement of Artificial Intelligence (AAAI)*, 2017.

J. Revaud, P. Weinzaepfel, Z. Harchaoui, and C. Schmid. Epicflow: Edge-preserving interpolation of correspondences for optical flow. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

J. Revaud, P. Weinzaepfel, Z. Harchaoui, and C. Schmid. Deepmatching: Hierarchical deformable dense matching. *International Journal of Computer Vision (IJCV)*, 2016.

J. Revaud, J. Almazán, R. S. Rezende, and C. R. d. Souza. Learning with average precision: Training image retrieval with a listwise loss. In *International Conference on Computer Vision (ICCV)*, 2019a.

J. Revaud, P. Weinzaepfel, C. R. de Souza, and M. Humenberger. R2d2: Repeatable and reliable detector and descriptor. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2019b.

E. Riba, D. Mishkin, D. Ponsa, E. Rublee, and G. Bradski. Kornia: an open source differentiable computer vision library for pytorch. In *Winter Conference on Applications of Computer Vision (WACV)*, 2020.

I. Rocco, R. Arandjelovic, and J. Sivic. Convolutional neural network architecture for geometric matching. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

I. Rocco, R. Arandjelović, and J. Sivic. End-to-end weakly-supervised semantic alignment. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018a.

I. Rocco, M. Cimpoi, R. Arandjelović, A. Torii, T. Pajdla, and J. Sivic. Neighbourhood consensus networks. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2018b.

I. Rocco, R. Arandjelović, and J. Sivic. Efficient neighbourhood consensus networks via submanifold sparse convolutions. In *European Conference on Computer Vision (ECCV)*, 2020.

O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer Assisted Intervention (MICCAI)*, 2015.

C. Rother, V. Kolmogorov, and A. Blake. "grabcut" interactive foreground extraction using iterated graph cuts. *ACM transactions on graphics (TOG)*, 2004.

C. Rother, T. Minka, A. Blake, and V. Kolmogorov. Cosegmentation of image pairs by histogram matching-incorporating a global constraint into mrfs. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2006.

M. Rubinstein, A. Joulin, J. Kopf, and C. Liu. Unsupervised joint object discovery and segmentation in internet images. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013.

B. C. Russell, A. Torralba, K. P. Murphy, and W. T. Freeman. Labelme: a database and web-based tool for image annotation. *International Journal of Computer Vision*, 2008.

K. Saenko, B. Kulis, M. Fritz, and T. Darrell. Adapting visual category models to new domains. In *European Conference on Computer Vision (ECCV)*, 2010.

B. Saleh and A. Elgammal. Large-scale classification of fine-art paintings: Learning the right metric on the right feature. *arXiv*, 2015.

B. Saleh, K. Abe, R. S. Arora, and A. Elgammal. Toward automated discovery of artistic influence. *Multimedia Tools and Applications*, 2016.

P.-E. Sarlin, D. DeTone, T. Malisiewicz, and A. Rabinovich. Superglue: Learning feature matching with graph neural networks. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

T. Sattler, T. Weyand, B. Leibe, and L. Kobbelt. Image retrieval for image-based localization revisited. In *British Machine Vision Conference (BMVC)*, 2012.

T. Sattler, W. Maddern, C. Toft, A. Torii, L. Hammarstrand, E. Stenborg, D. Safari, M. Okutomi, M. Pollefeys, J. Sivic, et al. Benchmarking 6dof outdoor visual localization in changing conditions. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

H. S. Sawhney. 3d geometry from planar parallax. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 1994.

J. L. Schonberger and J.-M. Frahm. Structure-from-motion revisited. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

J. L. Schonberger, F. Radenovic, O. Chum, and J.-M. Frahm. From single image query to detailed 3d reconstruction. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

B. Seguin, C. Striolo, F. Kaplan, et al. Visual link retrieval in a database of paintings. In *European Conference on Computer Vision (ECCV)*, 2016.

B. Seguin, I. diLenardo, and F. Kaplan. Tracking transmission of details in paintings. In *Digital Humanities (DH)*, 2017.

P. H. Seo, J. Lee, D. Jung, B. Han, and M. Cho. Attentive semantic alignment with offset-aware correlation kernels. In *European Conference on Computer Vision (ECCV)*, 2018.

X. Shen, A. A. Efros, and M. Aubry. Discovering visual patterns in art collections with spatially-consistent feature learning. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

X. Shen, F. Darmon, A. A. Efros, and M. Aubry. Ransac-flow: generic two-stage image alignment. In *European Conference on Computer Vision (ECCV)*, 2020a.

X. Shen, I. Pastrolin, O. Bounou, S. Gidaris, M. Smith, O. Poncet, and M. Aubry. Large-scale historical watermark recognition: dataset and a new consistency-based approach. In *International Conference on Pattern Recognition (ICPR)*, 2020b.

X. Shen, A. Joulin, A. A. Efros, and M. Aubry. Learning co-segmentation by segment swapping for retrieval and discovery. *arXiv*, 2021a.

X. Shen, Y. Xiao, S. X. Hu, O. Sbai, and M. Aubry. Re-ranking for image retrieval and transductive few-shot classification. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2021b.

X. Shen, R. Champenois, S. Ginosar, I. Pastrolin, M. Rousselot, O. Bounou, T. Monnier, S. Gidaris, F. Bougard, P.-G. Raverdy, M.-F. Limon, C. Bénévent, M. Smith, O. Poncet, K. Bender, J.-P. Béatrice, E. Honig, A. A. Efros, and M. Aubry. Spatially-consistent feature matching and learning for art collections and watermark recognition. *International Journal of Computer Vision (IJCV)*, 2022.

A. Shocher, N. Cohen, and M. Irani. "zero-shot" super-resolution using deep internal learning. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

A. Shrivastava, T. Malisiewicz, A. Gupta, and A. A. Efros. Data-driven visual similarity for cross-domain image matching. In *Proceedings of the 2011 SIGGRAPH Asia Conference*, 2011.

O. Siméoni, G. Puy, H. V. Vo, S. Roburin, S. Gidaris, A. Bursuc, P. Pérez, R. Marlet, and J. Ponce. Localizing objects with self-supervised transformers and no labels. In *British Machine Vision Conference (BMVC)*, 2021.

S. Singh, A. Gupta, and A. A. Efros. Unsupervised discovery of mid-level discriminative patches. In *European Conference on Computer Vision (ECCV)*, 2012.

J. Sivic and A. Zisserman. Video google: A text retrieval approach to object matching in videos. In *International Conference on Computer Vision (ICCV)*, 2003.

G. Strezoski and M. Worring. Omniart: multi-task deep learning for artistic data analysis. *arXiv*, 2017.

D. Sun, X. Yang, M.-Y. Liu, and J. Kautz. Pwc-net: Cnns for optical flow using pyramid, warping, and cost volume. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

J. Sun and J. Ponce. Learning dictionary of discriminative part detectors for image categorization and cosegmentation. *International Journal of Computer Vision (IJCV)*, 2016.

J. Sun, Z. Shen, Y. Wang, H. Bao, and X. Zhou. Loftr: Detector-free local feature matching with transformers. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

Y. Sun, X. Wang, Z. Liu, J. Miller, A. Efros, and M. Hardt. Test-time training with self-supervision for generalization under distribution shifts. In *International Conference on Machine Learning (ICML)*, 2020.

R. Szeliski. Image alignment and stitching: A tutorial. *Found. Trends. Comput. Graph. Vis.*, 2006.

W. R. Tan, C. S. Chan, H. E. Aguirre, and K. Tanaka. Ceci n'est pas une pipe: A deep convolutional network for fine-art paintings classification. In *International Conference on Image Processing (ICIP)*, 2016.

K. Tang, A. Joulin, L.-J. Li, and L. Fei-Fei. Co-localization in real-world images. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

T. Taniai, S. N. Sinha, and Y. Sato. Joint recovery of dense correspondence and cosegmentation in two images. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

B. Thomee, D. A. Shamma, G. Friedland, B. Elizalde, K. Ni, D. Poland, D. Borth, and L.-J. Li. Yfcc100m: The new data in multimedia research. *Communications of the ACM*, 2016.

Y. Tian, B. Fan, and F. Wu. L2-net: Deep learning of discriminative patch descriptor in euclidean space. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

G. Tolias, R. Sicre, and H. Jégou. Particular object retrieval with integral max-pooling of cnn activations. In *International Conference on Learning Representations (ICLR)*, 2015.

A. Torii, J. Sivic, T. Pajdla, and M. Okutomi. Visual place recognition with repetitive structures. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013.

A. Torii, R. Arandjelovic, J. Sivic, M. Okutomi, and T. Pajdla. 24/7 place recognition by view synthesis. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

P. Truong, M. Danelljan, L. Van Gool, and R. Timofte. Learning accurate dense correspondences and when to trust them. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021a.

P. Truong, M. Danelljan, F. Yu, and L. Van Gool. Warp consistency for unsupervised learning of dense correspondences. In *International Conference on Computer Vision (ICCV)*, 2021b.

I. Ubeda, J. M. Saavedra, S. Nicolas, C. Petitjean, and L. Heutte. Improving pattern spotting in historical documents using feature pyramid networks. *Pattern Recognition Letters*, 2020.

J. R. Uijlings, K. E. Van De Sande, T. Gevers, and A. W. Smeulders. Selective search for object recognition. *International Journal of Computer Vision (IJCV)*, 2013.

D. Ulyanov, A. Vedaldi, and V. Lempitsky. Deep image prior. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

N. Van Noord, E. Hendriks, and E. Postma. Toward discovery of the artist's style: Learning to recognize artists by their artworks. *IEEE Signal Processing Magazine*, 2015.

A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. Attention is all you need. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2017.

Y. Verdie, K. Yi, P. Fua, and V. Lepetit. Tilde: A temporally invariant learned detector. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

S. Vicente, C. Rother, and V. Kolmogorov. Object cosegmentation. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011.

H. V. Vo, F. Bach, M. Cho, K. Han, Y. LeCun, P. Pérez, and J. Ponce. Unsupervised image matching and object discovery as optimization. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

H. V. Vo, P. Pérez, and J. Ponce. Toward unsupervised, multi-object discovery in large-scale image collections. In *European Conference on Computer Vision (ECCV)*, 2020.

C. Vondrick, A. Shrivastava, A. Fathi, S. Guadarrama, and K. Murphy. Tracking emerges by colorizing videos. In *European Conference on Computer Vision (ECCV)*, 2018.

X. Wang, A. Jabri, and A. A. Efros. Learning correspondence from the cycle-consistency of time. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

Y. Wang, Y. Yang, Z. Yang, L. Zhao, P. Wang, and W. Xu. Occlusion aware unsupervised learning of optical flow. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018a.

Y. Wang, Y. Yang, Z. Yang, L. Zhao, P. Wang, and W. Xu. Occlusion aware unsupervised learning of optical flow. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018b.

Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *Transactions on Image Processing (TIP)*, 2004.

N. Westlake, H. Cai, and P. Hall. Detecting people in artwork with cnns. In *European Conference on Computer Vision (ECCV)*, 2016.

K. L. Wiggers, A. S. Britto, L. Heutte, A. L. Koerich, and L. S. Oliveira. Image retrieval and pattern spotting using siamese neural network. In *International Joint Conference on Neural Networks (IJCNN)*, 2019.

M. J. Wilber, C. Fang, H. Jin, A. Hertzmann, J. Collomosse, and S. J. Belongie. Bam! the behance artistic media dataset for recognition beyond photography. In *International Conference on Computer Vision (ICCV)*, 2017.

J. Wulff, L. Sevilla-Lara, and M. J. Black. Optical flow in mostly rigid scenes. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

R. Yin, E. Monson, E. Honig, I. Daubechies, and M. Maggioni. Object recognition in art drawings: Transfer of a neural network. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2016.

Z. Yin and J. Shi. Geonet: Unsupervised learning of dense depth, optical flow and camera pose. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

Y. Yuan, Y. Lyu, X. Shen, I. W. Tsang, and D.-Y. Yeung. Marginalized average attentional network for weakly-supervised learning. In *International Conference on Learning Representations (ICLR)*, 2019.

Z.-H. Yuan, T. Lu, Y. Wu, et al. Deep-dense conditional random fields for object co-segmentation. In *International Joint Conference on Artificial Intelligence (IJCAI)*, 2017.

J. Zhang, D. Sun, Z. Luo, A. Yao, L. Zhou, T. Shen, Y. Chen, L. Quan, and H. Liao. Learning two-view correspondences and geometry using order-aware network. *International Conference on Computer Vision (ICCV)*, 2019.

R. Zhang. Making convolutional networks shift-invariant again. In *International Conference on Machine Learning (ICML)*, 2019.

B. Zhou, H. Zhao, X. Puig, S. Fidler, A. Barriuso, and A. Torralba. Scene parsing through ade20k dataset. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

B. Zhou, H. Zhao, X. Puig, T. Xiao, S. Fidler, A. Barriuso, and A. Torralba. Semantic understanding of scenes through the ade20k dataset. *International Journal on Computer Vision (IJCV)*, 2018.

T. Zhou, Y. Jae Lee, S. X. Yu, and A. A. Efros. Flowweb: Joint image set alignment by weaving consistent, pixel-wise correspondences. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

T. Zhou, P. Krahenbuhl, M. Aubry, Q. Huang, and A. A. Efros. Learning dense correspondence via 3d-guided cycle consistency. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. *International Conference on Computer Vision (ICCV)*, 2017.