DISS. ETH NO. 23843

A DEEP UNDERSTANDING FROM A SINGLE IMAGE

A thesis submitted to attain the degree of

DOCTOR OF SCIENCES of ETH ZURICH
(Dr. sc. ETH Zurich)

presented by

RASMUS ROTHE

MEng Engineering Science,
University of Oxford
born on December 25, 1989
citizen of Germany

accepted on the recommendation of

Prof. Dr. Luc Van Gool, examiner
Prof. Dr. Nicu Sebe, co-examiner
Dr. Radu Timofte, co-examiner

2016

To my family.

ABSTRACT

As the number of photos being taken increases exponentially, there is also an exponentially increasing need for technologies for automatic image analysis. The applications include organizing personal photo collections, processing surveillance imagery, and industrial analysis. Despite the rise of deep learning in recent years, which brought computer vision to a level of maturity, there is still a long way to go until performance reaches the human level of image understanding: if you look at a photo, you will be immediately able to (a) sharpen the edges of the image in your mind even though it is slightly blurry, (b) recognize all objects in the image, (c) guess the age, gender, and attractiveness of the people, and (d) make a good guess about the occasion on which it was taken.

Can a computer make similar inferences? In this thesis we aim at providing answers to this question. We propose various techniques, including image processing, object detection, and fine-grained classification to infer as much as possible from just a single image.

As a first contribution, we propose an efficient novel artifact reduction algorithm based on an anchored regression model which doubles performance when compared to state-of-the-art methods while being orders of magnitude faster.

The second contribution is a novel formulation of non-maximum suppression (NMS) as a post processing step for object detection for a single image. Our method is based on the recent Affinity Propagation Clustering algorithm and contrary to the standard greedy approach solved globally with its parameters being learned automatically. The experiments show for object class and generic object detection that it provides a promising solution to the shortcomings of the greedy NMS.

The third contribution is a deep learning solution to age estimation from a single face image without the use of facial landmarks and the release of the IMDB-WIKI dataset, the largest public dataset of face images with age and gender labels. Our method

achieves state-of-the-art results for both real and apparent age estimation, winning an age estimation challenge against more than one hundred other competitors.

The fourth contribution is a framework to infer visual preferences from profile images and user ratings. Our computational pipeline comprises a face detector, convolutional neural networks for the extraction of deep features, a novel visual regularized collaborative filtering to infer inter-person preferences as well as a novel regression technique for handling visual queries without rating history. We validate the method using a very large dataset from a mobile dating application, images from celebrities as well as movie ratings and posters. We demonstrate our algorithms on howhot.io, which went viral around the Internet with more than fifty million pictures evaluated in the first month.

As the fifth contribution, we propose a framework for classifying cultural events from a single image. The method is based on extracting deep features at multiple scales, in combination with a sophisticated encoding and classification scheme. The proposed method is a top entry for a cultural event recognition challenge.

ZUSAMMENFASSUNG

Der exponentielle Anstieg an aufgenommenen Fotos erfordert auch eine exponentielle Weiterentwicklung der Technologien in der automatischen Bildverarbeitung. Die Anwendungen reichen vom Sortieren der privaten Fotosammlung, über die Auswertung von Überwachungskameras bis hin zu Applikationen in der Industrie. Obwohl der Erfolg von Deep Learning dazu geführt hat, dass die automatische Bildverarbeitung eine hohe Reife erlangt hat, ist die Leistungsfähigkeit in vielen Bereichen noch weit vom menschlichen Bildverständnis entfernt: wir Menschen können auf einem Foto sofort (a) verschwommene Ecken und Kanten unterbewusst schärfen, (b) alle Objekte auf dem Bild erkennen, (c) das Alter, Geschlecht und die Attraktivität von Personen einschätzen, (d) vermuten, zu welchem Anlass das Bild aufgenommen wurde.

Kann ein Computer vergleichbare Schlussfolgerungen treffen?

In dieser Dissertation versuchen wir, diese Frage zu beantworten. Dafür werden mehrere Techniken aus dem Bereich der Bildverarbeitung, Objekterkennung und der detailgenauen Bilderklassifizierung vorgestellt. Die Gemeinsamkeit dieser Ansätze ist, dass die Erkentnisse immer aus einem einzigen Foto abgeleitet werden.

Der erste Beitrag dieser Dissertation ist ein effizienter Algorithmus zur Reduktion von Bildkompressionsartefakten. Der Ansatz basiert auf einem neuartigen Regressionsverfahren, welches es nicht nur ermöglicht, die Qualität gegenüber bisherigen Methoden zu verdoppeln, sondern auch um Größenordnungen schneller ist.

Der zweite Beitrag ist eine neuartige Formulierung der Non-maximum Suppression (NMS) – ein wichtiger Nachbearbeitungsschritt in der Objekterkennung. Die vorgestellte Methode basiert auf dem neuartigen Affinity Propagation Clustering-Algorithmus und kann im Gegensatz zum gierigen NMS global gelöst werden, während die Hyperparameter automatisch gelernt werden können. Die präsentierten Experimente für generische und spezielle Objekterkennung zeigen, dass die neuartige Methode im Vergleich zum gierigen NMS eine Alternative mit großem Potenzial darstellt.

Als dritter Beitrag wird eine Deep Learning-basierte Lösung für die automatische Erkennung des Alters einer Person anhand eines einziges Bildes vorgestellt. Gleichzeitig wird der zur Zeit größte Datensatz (IMDB-WIKI) von mit Alter und Geschlecht annotierten Porträtbildern öffentlich verfügbar gemacht. Der vorgestellte Ansatz schlägt alle bisherigen Methoden in der Abschätzung von wahrem und wahrgenommenem Alter, und wurde bei einem Wettbewerb als die genauste von über hundert Methoden ausgezeichnet.

Der vierte Beitrag ist ein Framework, das es ermöglicht visuelle Präferenzen von Menschen anhand von Profilbildern und Bewertungen zu lernen und vorherzusagen. Der Ansatz besteht aus einer Gesichtserkennung, einem neuronalen Netzwerk für die extraktion von Features, einem neuartigen kollaborativen Filteralgorithmus, der visuell regularisiert wird, und es damit erlaubt, die Präferenzen zwischen Personen abzuleiten, sowie einer neuartigen Regressionstechnik, die einzig anhand von Profilbildern Präferenzen vorhersagt. Unsere Methode validieren wir mit einem großen Datensatz einer Mobile-Dating Applikation, mit Bildern von Prominenten sowie mit Postern und Bewertungen von Filmen. Weiterhin werden unsere Algorithmen auf der Website howhot.io verwendet, die weltweit viral ging, was dazu führte, dass im ersten Monat mehr als fünfzig Millionen Fotos hochgeladen wurden.

Der letzte Beitrag ist ein Framework, welches es erlaubt, sehr ähnlich aussehende kulturelle Veranstaltungen anhand eines einzigen Fotos voneinander zu unterscheiden. Die Methode extrahiert Deep Features auf mehreren Skalenebenen, gefolgt von einer anspruchsvollen Enkodierung und Klassifizierung.

ACKNOWLEDGMENTS

A PhD is like passing through a long tunnel: you enter highly motivated, in the middle barely see your own feet, while towards the end you are super thrilled to have started the journey when you begin to see the light at the end.

I am very grateful for all the support I received during this journey. This thesis is the result of several research collaborations, innumerable in-depth discussions and the general support of so many people.

First and foremost, I would like to express my gratitude to my advisor Luc Van Gool for letting me join his group. I cannot imagine a better supervisor – he gave me the freedom and trust to follow my own academic interests while at the same time offering close supervision when I needed it. Matthieu Guillaumin, thank you for bringing me up to speed and guiding me through my first year of the PhD. Thank you, Radu Timofte, for your close supervision: without your indestructible optimism, pragmatic but very analytical approach to difficult problems, and openness towards new ideas, I would never have reached the point at which I am now.

At the lab, I was surrounded by so many great people. Thank you all, in particular: Eirikur Agustsson and Marko Ristin for being great office mates with many long technical and non-technical discussions, Matthias Dantone for many fun weekends and a great conference trip to Japan, Valeria De Luca, and Tobias Gass for many refreshing coffee breaks, Lukas Bossard for all the technical support and discussions about Deep Learning, Andrea Fossati for making the most out of our joint project work, Angela Yao for welcoming me to the lab as my first office mate, Michael Gygli, Alex Locher, Hayko Riemenschneider and Anna Volokitin for a fun conference trip to the US, Nikolay Kobyshev for many long discus-

sions about starting a company, Santiago Manen, and Dengxin Dai for fun times during teaching, Nima Razavi for sharing the passion for entrepreneurship and helping to setup the alumni event of the lab, Peter Baki for the energizing workout sessions and last but not least Christina Krueger and Fiona Matthews for your endless support.

During my PhD, I supervised two semester projects. Thank you Prateek Purwar and Pushpak Pati for your great work.

Matthias Meier, thank you for proofreading my papers at any time of the day. Hanns Koenig, thanks for making a final check on the thesis draft.

I would also like to thank the Etzelschloss WG and the HackZurich team for making these 3 years in Zurich such an unforgettable time.

Most of all, I would like to thank my family: my mother, my father, my grandmother, my godfather and my two brothers.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

1

INTRODUCTION

In 2015 more than 1 trillion photos were taken globally [1]. Printed in standard size and lined up next to each other, they would stretch out for more than a round trip from the Earth to the Sun. As this number increases exponentially from year to year, there is also an exponentially increasing need for technologies that are capable of analyzing images automatically. This includes applications which organize personal photo collections, processing surveillance imagery, and industrial applications. With the rise of deep learning in recent years [77], [89], [117], computer vision solutions have finally reached a level of maturity that enables them to be applied to real-world problems. Nonetheless, there is still a long way to go until these solutions reach the human level of understanding: if you look at the photo in Figure 1.1, you will be immediately able to (a) sharpen the edges of the image in your mind even though it is slightly blurry, (b) recognize all objects in the image, (c) guess the age, gender, and attractiveness of the four people, and (d) know that it was taken on St. Patrick's Day.

Can a computer come to a similar conclusion? In this thesis we aim at providing answers to this question. We base our predictions always on a single image, as in real world scenarios we often do not have the luxury of having multiple images of the same object, person or scene. This can be difficult when the image was captured at a low resolution, was compressed, objects are occluded or lighting is not optimal. However, even when the object of interest is fully visible and can be correctly classified, automatic fine-grained classification such as inferring biometrics like age, gender, or attractiveness or event detection are still difficult when judging from a single image.

Thus the overarching goal of this thesis is to propose various techniques from image processing, over detection to fine-grained classification to be able to infer as much as possible from a single

[1] http://mylio.com/true-stories/next/one-trillion-photos-in-2015-2

Figure 1.1: What can we infer from a single image?

image. We start with the raw image by proposing a super resolution technique to reduce image compression artifacts to recover as much details as possible. After objects have been detected by a detector, our novel non-maximum suppression scheme allows to recover as many detections as possible from a single image. After all objects in an image have been successfully detected, in a next step we focus on fine-grained classification of biometrics such as age, gender and facial beauty prediction using deep learning. For the latter, we present a framework to predict personalized visual preferences. The thesis is concluded by applying a sophisticated

pipeline for fine-grained event classification. Figure 1.1 summarizes the proposed methods.

1.1 CONTRIBUTIONS

Specific contributions of this thesis are listed as follows:

- The first contribution is an efficient novel artifact reduction algorithm based on the adjusted anchored neighborhood regression (A+) [141]. The proposed method doubles the relative gains in PSNR when compared to state-of-the-art methods such as Semi-local Gaussian Processes (SLGP) [82], while being order(s) of magnitude faster.

- The second contribution is a novel formulation of non-maximum suppression (NMS) as a post-processing step for object detection for a single image. Our method is based on the recent Affinity Propagation Clustering algorithm [44] and, contrary to the standard greedy approach, solved globally with its parameters being learned automatically. The experiments show for object class and generic object detection that it provides a promising solution to the shortcomings of the greedy NMS.

- The third contribution is a deep learning solution to age estimation from a single face image without the use of facial landmarks and the release of the IMDB-WIKI dataset, the largest public dataset of face images with age and gender labels. Our method achieves state-of-the-art results for both real and apparent age estimation, winning the Chalearn Looking at People (LAP) age estimation challenge [38] against 115 other competitors.

- The fourth contribution is a framework to infer visual preferences from profile images and user ratings. Our computational pipeline comprises a face detector, convolutional neural networks for the extraction of deep features, a novel visual regularized collaborative filtering to infer inter-person preferences as well as a novel regression technique for handling visual queries without rating history. We validate the

method using a very large dataset from a dating site, images from celebrities as well as on the standard MovieLens rating dataset, augmented with movie posters. We demonstrate our algorithms on www.howhot.io which went viral around the Internet, with more than 50 million pictures evaluated in the first month.

- The fifth contribution is a framework for classifying cultural events from a single image. The method is based on extracting deep features at multiple scales, which are then encoded using Linear Discriminant Analysis (LDA) and classified through the Iterative Nearest Neighbors-based classifier (INNC) [146]. The proposed method is a top entry for the ChaLearn LAP 2015 cultural event recognition challenge [38].

1.2 PUBLICATIONS

This thesis contains and discusses extended or modified versions of the following publications:

- R. Rothe, R. Timofte, and L. Van Gool. Efficient regression priors for reducing image compression artifacts. In *IEEE International Conference on Image Processing*, 2015. [127]
- R. Rothe, M. Guillaumin, and L. Van Gool. Non-Maximum Suppression for Object Detection by Passing Messages between Windows.
  In *Asian Conference on Computer Vision*, 2014. [122]
- R. Rothe, R. Timofte, and L. Van Gool. Deep expectation of real and apparent age from a single image without facial landmarks. *International Journal of Computer Vision*, 2016. [124]
- R. Rothe, R. Timofte, and L. Van Gool. Some like it hot - visual guidance for preference prediction. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2016. [128]
- R. Rothe, R. Timofte, and L. Van Gool. DLDR: Deep Linear Discriminative Retrieval for cultural event classification from a single image. In *IEEE International Conference on Computer Vision Workshops*, 2015. [126]

During the time of this Ph.D., research on various related topics to this thesis was conducted. This includes further work on single image super resolution techniques, age, gender and other attribute estimation from a single image as well as apparel classification. These works are not discussed in this thesis, but listed here for the sake of completeness:

- R. Rothe, M. Ristin, M. Dantone, and L. Van Gool. Discriminative Learning of Apparel Features. In *Conference on Machine Vision Applications*, 2015. [123]
- R. Rothe, R. Timofte, and L. Van Gool. DEX: Deep EXpectation of apparent age from a single image. In *IEEE International Conference on Computer Vision Workshops*, 2015. [125]
- R. Timofte, R. Rothe, and L. Van Gool. Seven ways to improve example-based single image super resolution. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2016. [144]
- M. Uricar, R. Timofte, R. Rothe, J. Matas, and L. Van Gool. Structured Output SVM Prediction of Apparent Age, Gender and Smile From Deep Features. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2016. [151]
- R. Torfason, E. Agustsson, R. Rothe, and R. Timofte. From face images and attributes to attributes. In *Asian Conference on Computer Vision*, 2016. [148]

1.3 ORGANIZATION

This thesis is organized as follows:

In **Chapter 2**, *Reducing image compression artifacts*, we propose a novel method for suppressing compression artifacts from a single image. Very recently, a learned semi-local Gaussian Processes-based solution (SLGP) [82] has been proposed with impressive results. However, when applied to top compression schemes such as JPEG 2000, the improvement is less significant. We propose an efficient novel artifact reduction algorithm based on the adjusted anchored neighborhood regression (A+) [141], a method

from the super-resolution literature. We double the relative gains in peak signal-to-noise ratio (PSNR) when compared to state-of-the-art methods such as SLGP, while being order(s) of magnitude faster.
This chapter is based on research originally presented in Rothe *et al.*, *IEEE International Conference on Image Processing, 2015* [127].

In **Chapter 3**, *Non-maximum suppression for object detection*, we propose a novel formulation of non-maximum suppression as a post processing step for object detection for a single image. Non-maximum suppression (NMS) is a key post-processing step in many computer vision applications. In the context of object detection, it is used to transform a smooth response map that triggers many imprecise object window hypotheses in, ideally, a single bounding-box for each detected object. The most common approach for NMS for object detection is a greedy, locally optimal strategy with several manually-designed components. Such a strategy inherently suffers from several shortcomings, such as the inability to detect nearby objects. In this chapter, we try to alleviate these problems and explore a novel formulation of NMS as a well-defined clustering problem. Our method builds on the recent Affinity Propagation Clustering algorithm [44], which passes messages between data points to identify cluster exemplars. Contrary to the greedy approach, our method is solved globally and its parameters can be automatically learned from training data. In experiments, we show in two contexts – object class and generic object detection – that it provides a promising solution to the shortcomings of the greedy NMS. This chapter is based on research originally presented in Rothe *et al.*, *Asian Conference on Computer Vision, 2014* [122].

In **Chapter 4**, *Predicting real and apparent age*, we propose a deep learning solution to age estimation from a single face image without the use of facial landmarks and introduce the IMDB-WIKI dataset, the largest public dataset of face images with age and gender labels. While real age estimation research spans over decades, the study of apparent age estimation or of age as perceived by other humans from a face image is a recent endeavor. We tackle both tasks with our convolutional neural networks (CNNs) of VGG-

16 architecture [134] which are pre-trained on ImageNet for image classification. We pose the age estimation problem as a deep classification problem followed by a softmax expected value refinement. The key factors of our solution are: deep learned models from large data, robust face alignment, and expected value formulation for age regression. We validate our methods on standard benchmarks and achieve state-of-the-art results for both real and apparent age estimation. This chapter is based on research originally presented in Rothe *et al.*, *International Journal of Computer Vision, 2016* [124].

In **Chapter 5**, *Visual guidance for preference prediction*, we present a framework to infer visual preferences from profile images and user ratings. For people, first impressions of someone are of determining importance. They are hard to alter through further information. This begs the question whether a computer can reach the same judgment. Earlier research has already pointed out that average attractiveness can be estimated with reasonable precision. We improve the state of the art, but also predict – based on someone's known preferences – how much that particular person is attracted to a novel face. Our computational pipeline comprises a face detector, convolutional neural networks for the extraction of deep features, standard support vector regression for facial beauty prediction, and – as the main novelties - visual regularized collaborative filtering to infer inter-person preferences as well as a novel regression technique for handling visual queries without rating history. We validate the method using a very large dataset from a dating site as well as images from celebrities. Our experiments yield convincing results, *i.e.* we predict 76% of the ratings correctly solely based on an image, and reveal some sociologically relevant conclusions. We also validate our collaborative filtering solution on the standard MovieLens rating dataset, augmented with movie posters, to predict an individual's movie rating. We demonstrate our algorithms on `www.howhot.io` which went viral around the Internet with more than 50 million pictures evaluated in the first month. This chapter is based on research originally presented in Rothe *et al.*, *IEEE Conference on Computer Vision and Pattern Recognition, 2016* [128].

In **Chapter 6**, *Deep retrieval for cultural event classification*, we tackle the classification of cultural events from a single image with a deep learning-based method. We use convolutional neural networks (CNNs) with VGG-16 architecture [134], pretrained on ImageNet or the Places205 dataset for image classification, and finetuned on cultural events data. Deep features are robustly extracted at four different layers in each image. At each layer, Linear Discriminant Analysis (LDA) is employed for discriminative dimensionality reduction. An image is represented by the concatenated LDA-projected features from all layers or by the concatenation of CNN pooled features at each layer. The classification is then performed through the Iterative Nearest Neighbors-based Classifier (INNC) [146]. Classification scores are obtained for different image representation setups at train and test. The average of the scores is the output of our deep linear discriminative retrieval (DLDR) system. With 0.80 mean average precision (mAP), DLDR is a top entry for the ChaLearn LAP 2015 cultural event recognition challenge [38]. This chapter is based on research originally presented in Rothe *et al.*, *IEEE International Conference on Computer Vision Workshops, 2015* [126].

In **Chapter 7** we conclude the thesis by pointing out the main contributions and proposing future work.

2

REDUCING IMAGE COMPRESSION ARTIFACTS

2.1 INTRODUCTION

For the sake of reducing storage, images are often is stored in compressed form. Furthermore, lossy image compression is preferred to lossless compression because of its significantly higher compression rates. This, however, results in the loss of fidelity to the original content. With the broad adoption of lossy image compression, in particular the compression artifact suppression has become a focus for research. Thus as a first contribution of this thesis we propose a method to reduce image compression artifacts to recover as much details as possible from a single image. Besides improving the quality of the image for humans looking at it, Dai *et al*. [26] showed that removing image artifacts can also help for object detection as discussed in Chapter 3 as well as for fine-grained classification as discussed in Chapter 4, 5, and 6.

The related literature closely connects with important advances that have been made in compression algorithms.

One of the most used coding techniques is block-based discrete cosine transform (BDCT). It is used widely for compression of both images and videos (e.g., JPEG/MPEG). BDCT's main drawback is the presence of discontinuities at block boundaries, also known as block artifacts, especially for low bit rates. JPEG 2000 uses the discrete wavelet transform instead of the BDCT stage from JPEG. In this way, JPEG 2000 exhibits milder artifacts, mostly ringing artifacts. In [107] a scale-space method for ringing estimation is proposed.

There are different research directions for artifact removal seen as an image enhancement task. The main one, followed in this chapter, employs the use of prior knowledge. Projection onto convex sets (POCS) models represent prior knowledge under the form of convex constraints (such as smoothness or quantization constraints). POCS models perform well for JPEG [172] and JPEG

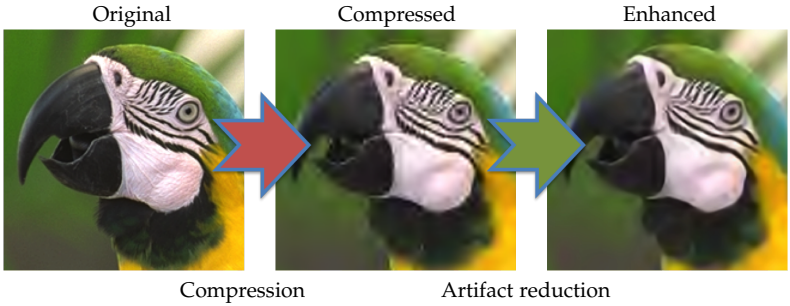


Figure 2.1: Image compression artifact reduction result of our method (image 6).

2000 [94] image enhancement. Roth and Black [121] propose a field of experts (FoE) with learned clique potentials under Markov Random Field framework for image enhancement. The noise removal is targeted by Laparra *et al*. [85] with a non-parametric Support Vector Regression (SVR) method. Tschumperle and Deriche [149] propose a single generic anisotropic diffusion equation as unifying expression for different enhancement applications.

Other works propose specific formulations for the compression artifact removal. Qiu [113] proposes the use of a multi-layer perceptron (MLP) model. Foi *et al*. [43] applies a shape-adaptive DCT method (SADCT) pointwisely. Zhai *et al*. [181] uses a block-shift filtering-based algorithm.

Nosratinia [109] observes that the re-application of JPEG reduces the artifacts. He notices the same for JPEG 2000 [110].

Recently, Kwon *et al*. [82] proposed a common solution to image super-resolution and compression artifact removal by using Gaussian Processes (GP) under a semi-local approximation (SLGP). By the approximation scheme the time complexity of large-scale GPs decreases. Since their approach achieves the best results to date for JPEG and JPEG 2000 artifact removal, it is our main comparing method.

We propose a novel post-processing method based on the recent adjusted anchored neighborhood regression (A+) [141], [165], a state-of-the-art method in single image super-resolution. In our method, for a certain lossy compression method we learn offline

linear regressors from compressed to raw train images, and then apply them to reduce the compression artifacts in test images. Based on these priors extracted from the training material we are able to reduce the artifacts and achieve state-of-the-art performance, doubling the PSNR gain of SLGP [82] while having order of magnitude lower running time.

2.2 PROPOSED METHOD

2.2.1 *Overview*

Our method follows closely the adjusted anchored neighborhood regression (A+) super-resolution method of Timofte *et al*. [141]. The method works with small image patches of fixed size (e.g. 7×7 pixels). The patches are extracted densely over an image grid. The offline training starts with the extraction of pairs of patches in the training compressed image (low resolution, LR) and the corresponding ones in the raw artifact-free image (high resolution, HR). The patches are used to train a sparse dictionary whose atoms/patches are taken as representatives of the compressed image space. These are the anchoring points of our method. For each we offline compute a regressor from the compressed to the artifact-free image patches (from LR to HR). At test time, we extract over a grid patches and regress them to the artifact-free image by picking the stored regressor at its nearest anchoring point. The regressed patches are averaged in the overlapped areas to form the output enhanced image.

2.2.2 *Patches and features*

The LR patches are represented by their features which are concatenated responses to 1st and 2nd order gradients applied horizontally and vertically to the LR image. We use the same features as in [141], [142], [180]. Through PCA we reduce the dimensionality of the features, while keeping 99.9% of the energy. The HR patches are represented by the difference between the ground truth HR image and the LR image. The training LR features are normalized by

l_2-norm and the corresponding HR patches are scaled accordingly, as in [141]. At test time, the reconstructed image is added to the input LR image for the final output.

2.2.3 *Anchoring points*

The relation between the patches from compressed images (LR) and their corresponding artifact-free images (HR) is heavily non-linear. Instead of training a single non-linear regression function to model this complex relation, we partition the LR space around anchoring points and train local linear regressors to the HR space as in [141], [142], which results in a very good approximation.

In order to obtain the anchoring points in LR space, a dictionary $\mathbf{D}_l$, we use the K-SVD [1] method, as in [141], [180].

$$\begin{aligned} \mathbf{D}_l, \{\mathbf{w}^{(k)}\} = \quad & \operatorname{argmin}_{\mathbf{D}_l, \{\mathbf{w}^{(k)}\}} \textstyle\sum_k \|\mathbf{p}_l^{(k)} - \mathbf{D}_l \mathbf{w}^{(k)}\|^2 \\ & \text{s.t. } \|\mathbf{w}^{(k)}\|_0 \leq L \; \forall k, \end{aligned} \tag{2.1}$$

where $\mathbf{p}_l^{(k)}$ are the training LR features, L is the imposed sparsity, fixed to 3, and $\mathbf{w}^{(k)}$ are the decomposition coefficients over $\mathbf{D}_l$.

2.2.4 *Anchored regressors*

We train a linear regressor locally for each anchoring point by solving, as in [141]:

$$\min_{\boldsymbol{\beta}} \{\|\mathbf{y} - \mathbf{S}_{l,\mathbf{d}_\mathbf{y}} \boldsymbol{\beta}\|^2 + \lambda \|\boldsymbol{\beta}\|^2\}, \tag{2.2}$$

where $\mathbf{y}$ is a LR patch feature whose nearest anchoring point is $\mathbf{d}_\mathbf{y} \in \mathbf{D}_l$, $\mathbf{S}_{l,\mathbf{d}_\mathbf{y}}$ are the N nearest neighbors in the training pool for $\mathbf{d}_\mathbf{y}$, and λ is the regularization parameter, here fixed to 0.1. The regressor $\mathbf{P}_{\mathbf{d}_\mathbf{y}}$ corresponding to the anchoring point $\mathbf{d}_\mathbf{y}$ is computed offline:

$$\mathbf{P}_{\mathbf{d}_\mathbf{y}} = \mathbf{S}_{h,\mathbf{d}_\mathbf{y}} (\mathbf{S}_{l,\mathbf{d}_\mathbf{y}}^T \mathbf{S}_{l,\mathbf{d}_\mathbf{y}} + \lambda \mathbf{I})^{-1} \mathbf{S}_{l,\mathbf{d}_\mathbf{y}}^T, \tag{2.3}$$

where $\mathbf{S}_{h,\mathbf{d}_\mathbf{y}}$ contains the HR patches corresponding to the LR vectors in $\mathbf{S}_{l,\mathbf{d}_\mathbf{y}}$.

Figure 2.2: Evaluation dataset with 16 images *aka* DB1 [82]. The images are numbered 1-8 on 1^{st} row and 9-16 on 2^{nd} row.

2.2.5 *Runtime*

At test time, we first extract from the input compressed image the patch features densely over a grid. For each input LR feature $\mathbf{y}$ we retrieve the nearest neighboring anchoring point $\mathbf{d_y} \in \mathbf{D}_l$ and obtain the output $\mathbf{x}$ by applying the stored regressor $\mathbf{P}_{l,\mathbf{d_y}}$ at anchoring point $\mathbf{d_y}$:

$$\mathbf{x} = \mathbf{P}_{l,\mathbf{d_y}}\mathbf{y} \tag{2.4}$$

The regressed $\{\mathbf{x}\}$ patches are averaged in the overlapping areas to obtain the correcting output image. Finally, the input LR image is added to obtain the complete enhanced output HR image.

2.3 EXPERIMENTS

In this section we evaluate the performance of our proposed method. We show how its performance is influenced by the design parameters and compare to state-of-the-art methods on a standard dataset.

2.3.1 *Benchmark*

For a fair comparison with SLGP we use the same images for testing as Kwon *et al*. [82]. The dataset (see Fig. 2.2) contains 16 images familiar to the community (512×512 or 256×256 pixels). While Kwon *et al*. [82] uses 500 training images from a personal collection, we use the training set of 91 images proposed by Yang *et al*. [170] and extract 5 million patches from them by computing first image

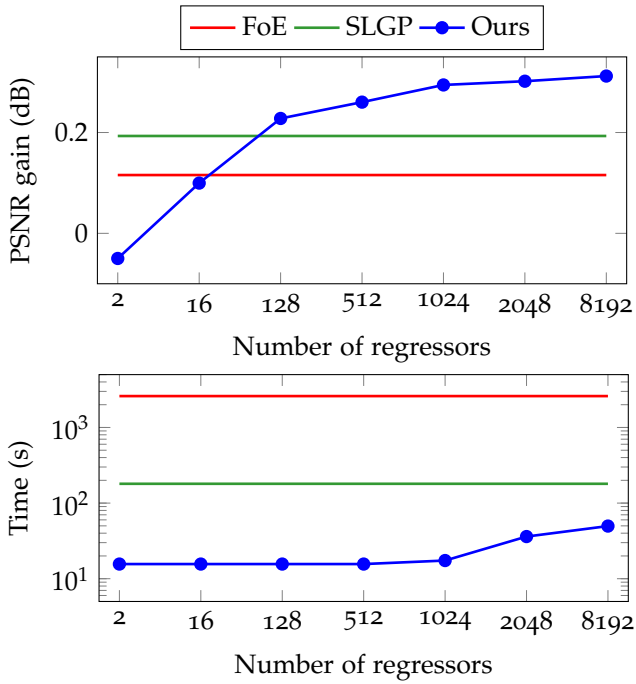


Figure 2.3: Number of regressors vs. performance and running time.

scale pyramids with downscaling factor 0.98 and bicubic interpolation.

We compare our method with the re-application of JPEG 2000 method [110], Field of Experts (FoE) [121] and Semi-local Gaussian Processes (SLGP) [82] (state-of-the-art).

Each image was degraded using the JPEG 2000 encoder from the Kakadu software package [1] at 0.1 bits per pixel (BPP) at test time, a compression at which the artifacts are usually easily noticed. At training time the images are compressed at only 0.3 BPP. At this lower compression rate the regressors can more easily pick up the patterns of the artifacts which leads to an improvement in performance. The performance of the enhancement methods is

[1] http://www.kakadusoftware.com

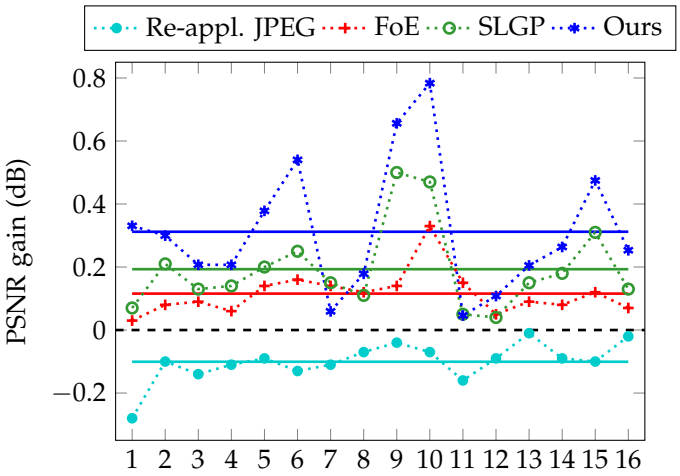
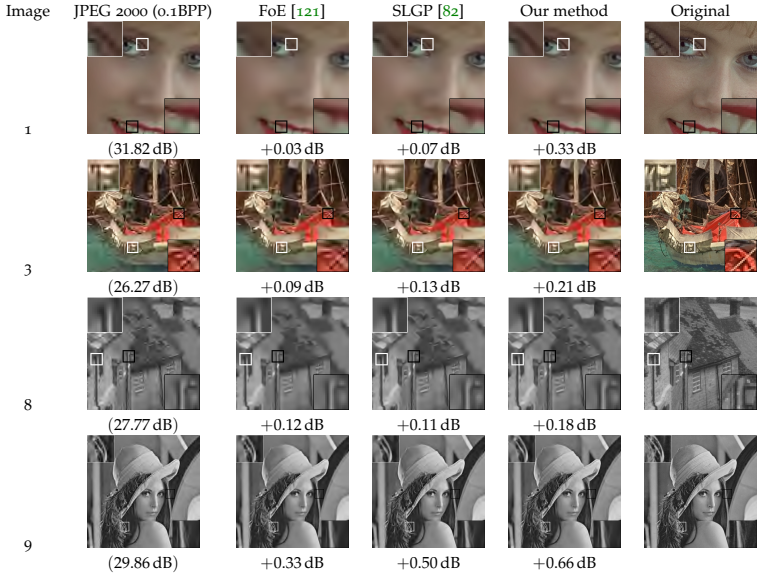


Figure 2.4: PSNR gain comparison of the proposed method against re-application of JPEG 2000, FoE, and SLGP image enhancement algorithms. The x axis corresponds to the image index as in Fig. 2.2. The average PSNR gains across the dataset are marked with solid lines.

measured by evaluating the peak signal-to-noise ratio (PSNR) to the uncompressed image. We report the PSNR gain relatively to the degraded image. Note that for our method we use the YCbCr color space for the color images (10 out of 16 images) and perform the enhancement only on the Y channel.

Our choice to work directly with JPEG 2000 is due to the increased difficulty in obtaining significant improvements from most current artifact reduction methods (often less than 0.1dB for 0.1 BPP). With respect to JPEG, JPEG 2000 is a superior compression algorithm, provides better quality for the same BPP. Also JPEG exhibits stronger artifacts, partly due to the BDCT stage (block artifacts), and it is easier to enhance (often over 0.5dB for 0.1 BPP).

Figure 2.5: Qualitative results for image 1, 3, 8, and 9 from the testing dataset (see Fig. 2.2). Best seen on screen.



2.3.2 *Parameters*

The default main parameters of our method are: 5 million training pairs of LR and HR patches, 7×7 pixels patch size, 2048 anchoring points / regressors, and 2048 nearest neighbors for the offline computation of each regressor.

For the A+ method [141] applied to super-resolution it was shown that increasing the number of training patches leads to increased in PSNR performance, and indeed our preliminary experiments confirmed the same behavior for our method on the artifact reduction task.

For the patch size we considered 3×3, 5×5, 7×7, and 9×9 patch sizes. The performance improved up to 7×7, but slightly diminished for 9×9. Therefore, our choice of patch size (7×7) matches the one from the SLGP method [82]. Note that at 3×3

patches and 1024 regressors our method still gains 0.194dB, comparable to SLGP with 7×7 patches.

The number of linear regressors / anchoring points (dictionary size) is evaluated in Fig. 2.3 with respect to PSNR gain and average running time per image. There is a linear relation between the number of regressors and the running time, since a linear search is involved for picking up the nearest anchoring point and stored regressor for each input patch. The linearity holds above 512 regressors when the searching time dominates. Our method is order(s) of magnitude faster than the compared FoE and SLGP methods. With as few as 16 regressors our method reaches the PSNR gain of the FoE method and with 128 regressors clearly outperforms the SLGP method. Our method peaks at 0.312dB for 8192 regressors, but reaches a plateau at 1024 regressors (0.302dB). We expect that by increasing the size of the training set of images and its variance, as well as potentially the number of training patches, the performance of our method could be even more improved. It might at this point also be noted that in the current setting we only use 91 images, while SLGP uses 500 images. Our method is well behaved: more training data or more regressors usually results in better performance.

2.3.3 *Performance*

In order to assess the performance of our method we build up our benchmark following the settings from Kwon *et al*. [82] as used for their SLGP state-of-the-art method. In Fig. 2.4 we compare the proposed method against re-application of JPEG 2000 method, FoE, and SLGP in terms of PSNR gain. We keep the same image indices from [82], as depicted in Fig. 2.2 and report also the average performance. Our method improves over SLGP for all the images, except image 7. Also in average performance we achieve a strong 0.312dB, significantly better than SLGP with 0.192dB and FoE with 0.115dB. The re-application of JPEG 2000 leads to negative gains.

The running time of our method compares favorable with the other top methods such as SLGP or FoE (see Fig. 2.3). If SLGP requires 180s and FoE $\sim$ 2600s per 512×512 pixels images, our method needs only 15s with 1024 regressors (Matlab). Our codes

are publicly available at:
http://www.vision.ee.ethz.ch/~timofter/

For the qualitative performance assessment we compare enhancement results for 4 images in Fig. 2.5. We notice a clear improvement in quality between the JPEG 2000 input image and the result of SLGP or of our method. FoE tends to oversmooth the edges, while our method produces relatively sharp edges and remains closer to the uncompressed original image.

Our method improves $\sim$ 0.12dB over the SLGP method. While this might be a small improvement in absolute terms, it is a very solid result given the difficult scenario (JPEG 2000 @ 0.1BPP) we dealt with. In fact, looking at relative terms, our method with 0.31dB gain almost doubles the performance of SLGP (0.19dB), and triples FoE (0.11dB). Moreover, our method is orders of magnitude faster.

2.4 CONCLUSION

In this chapter, we propose a novel and efficient artifact reduction algorithm based on A+. We embed prior information from the training images and the compressed outputs into a set of learned linear regressors. At test, after applying these we improve the compressed images by reducing the artifacts. The experiments show large improvements doubling the PSNR gain when compared to state-of-the-art methods such as SLGP [82], while being an order of magnitude faster. In the next chapter we propose a novel non-maximum supression scheme to improve the detection of objects from a single image. Dai *et al*. [26] showed that applying artifact suppression in the form of super resolution helps to improve the detection accuracy. Thus the method proposed in this chapter can be seen as a preprocessing step for the detection stage presented in the following Chapter 3. This can become relevant especially for later fine-grained classifaction of the objects (*i.e*. age, attractiveness or events, see Chapter 4, 5, and 6), where subtle details can make a large difference.

3 NON-MAXIMUM SUPPRESSION FOR OBJECT DETECTION

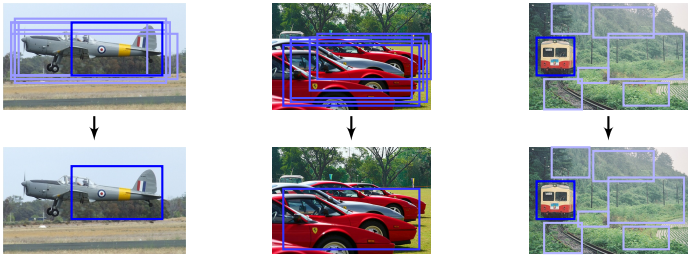
3.1 INTRODUCTION

The goal of this chapter of the thesis is to present a novel non-maximum suppression (NMS) scheme aiming at improving the detection of objects. This detection pipeline can be seen as a preprocessing step to the fine-grained classification algorithms presented in the next two chapters.

Non-maximum suppression has been widely used in several key aspects of computer vision and is an integral part of many proposed approaches in detection, might it be edge, corner or object detection [14], [21], [28], [42], [52], [155]. Its necessity stems from the imperfect ability of detection algorithms to localize the concept of interest, resulting in groups of several detections near the real location.

In the context of object detection, approaches based on sliding windows [28], [42], [155] typically produce multiple windows with high scores close to the correct location of objects. This is a consequence of the generalization ability of object detectors, the smoothness of the response function and visual correlation of close-by windows. This relatively dense output is generally not satisfying for understanding the content of an image. As a matter of fact, the number of window hypotheses at this step is simply uncorrelated with the real number of objects in the image. The goal of NMS is therefore to retain only one window per group, corresponding to the precise local maximum of the response function, ideally obtaining only one detection per object. Consequently, NMS also has a large positive impact on performance measures that penalize double detections [2], [39].

The most common approach for NMS consists of a greedy iterative procedure [28], [42], which we refer to as *Greedy NMS*. The procedure starts by selecting the best scoring window and assum-



(a) The top-scoring box may not be the best fit.

(b) It may suppress nearby objects.

(c) It does not suppress false positives.

Figure 3.1: Examples of possible failures when using a greedy procedure for NMS.

ing that it indeed covers an object. Then, the windows that are too close to the selected window are suppressed. Out of the remaining windows, the next top-scoring one is selected, and the procedure is repeated until no more windows remain. This procedure involves defining a measure of similarity between windows and setting a threshold for suppression. These definitions vary substantially from one work to another, but typically they are manually designed. Greedy NMS, although relatively fast, has a number of downsides, as illustrated in Fig. 3.1. First, by suppressing everything within the neighborhood with a lower confidence, if two or more objects are close to each other, all but one of them will be suppressed. Second, Greedy NMS always keeps the detection with the highest confidence even though in some cases another detection in the surrounding might provide a better fit for the true object. Third, it returns all the bounding-boxes which are not suppressed, even though many could be ignored due to a relatively low confidence or the fact that they are sparse in a subregion within the image.

As these problems are due to greediness and hard-thresholding, in this chapter we propose to consider NMS as a clustering problem that is solved globally, where the hard decisions taken by Greedy NMS are replaced with soft penalties in the objective function. The intuition behind our model is that the multiple proposals for the same object should be grouped together and be represented by just one window, the so-called *cluster exemplar*. We

therefore adopt the framework of Affinity Propagation Clustering (APC) [44], an exemplar-based clustering algorithm, which is inferred globally by passing messages between data points.

However, APC is not directly usable for NMS. We need to adapt it to include two constraints that are specific to detection. First, since there are false positives, not every window has to be assigned to a cluster. Second, in certain scenarios it is beneficial to encourage a diverse set of proposals and penalize selecting exemplars that are very close to each other. Hence, our contributions are the following: (i) we extend APC to add repellence between cluster centers; (ii) to model false positives, we relax the clustering problem; (iii) we introduce weights between the terms in APC, and show how these weights can be learned from training data.

We show in our experiments that our approach helps to address the limitations of Greedy NMS in two different contexts: object class detection (Sec. 3.4) and generic object detection (Sec. 3.5).

3.2 RELATED WORK

NMS is a widely used post-processing technique in several computer vision applications. For edge, corner and interest point detection, its role is to find the local maxima of a function defined over a pixel scale-space pyramid, and it is common to simply suppress any pixel which is not the maximum response in its neighborhood [14], [104].

Similarly, for object detection, many approaches have been proposed to prune the set of responses that score above the detection threshold. The Viola-Jones detector [155] partitions those responses in disjoint sets, grouping together responses as soon as they overlap, and propose, for each group with enough windows, a window whose coordinates are the group average. Recently, a more common approach has been to adopt a greedy procedure [28], [42], [132] where the top-scoring window is declared an object, then neighboring windows are removed based on a hand-tuned threshold of a manually-designed similarity (distance between centers when the size ratio is within $0.5 - 2$ in [28], [132]; relative size of the intersection of the windows with respect to the selected object window in [42]). Most current object category detection

pipelines [22], [138], [150], but also generic object detection ones [2], use such a greedy procedure. As explained in the introduction, a greedy approach with manually-set parameters is not fully satisfactory.

Several alternatives have been considered. A first line of work considers the detector response as a distribution, and formulates the goal of NMS as that of finding the modes of this distribution. For instance, mean-shift for a kernel density estimation [27] and mixtures of scale-sensitive Gaussians [163] have been proposed. Although principled, these approaches still select only local maxima and fail to suppress false positive detections.

A second line of approaches includes iterative procedures to progressively remove extraneous windows. In [9], a re-ranking cascade model is proposed where a standard greedy NMS is used at every step to favor sparse responses. In [18], the authors also adopt an iterative procedure. From a base detector model, a more powerful detector is built using local binary patterns that encode the neighborhood of window scores in the target image. The procedure is iterated several times until saturation of the detector. This is very similar to the idea of contextual boosting [31]. These iterative procedures are rather time-consuming, as they involve re-training object detectors at each iteration.

For the special case of object detection performed through voting, NMS can be done implicitly by preventing a vote to be taken multiple times into account. For instance, with Hough Forests [5], [115], [162], patches vote for the location of the object center. The location with maximum response is selected as the object, and the votes within a given radius that contribute to the selected center are removed from the Hough space hence preventing double detections.

The same idea applies to part-based voting for detection [164]. However, these approaches are not generic and do not apply to every object detection framework. In [8], [10], the authors propose to include repulsive pairwise terms into the search for high-scoring windows, so as to avoid performing NMS as a post-processing step. The search is performed using branch-and-bound techniques.

As mentioned earlier, Greedy NMS has the potential shortcoming of suppressing occluding or nearby instances. Several works

aim at solving this problem in particular. For the problem of pedestrian detection, [140] proposed to learn detection models for couples of person. Unfortunately, this idea scales very unfavorably with the number of elements in a group, and creates new problems for NMS: what should be done when a double-detection and two single detections are found nearby?

A related field of research generalizes the idea of NMS to the problem of detecting multiple object classes at the same time. This is often referred to as *context rescoring* [29], [42]. Those approaches explicitly model co-occurrence and mutual exclusion of certain object classes, and can incorporate NMS and counts for a given object class [29]. Several works go even further and also model scene type and pixel-level segmentation jointly [83], [174].

To the best of our knowledge, our work is the first to view NMS as a message-passing clustering problem. Clustering algorithms like k-means [100], k-medoids [73] and spectral clustering [156] are not well suited because they return a fixed number of clusters. However, the number of objects and therefore ideal number of clusters is an unknown prior and thus should not have to be fixed in advance. This inflexibility results in poor performance as shown in the experiments. We overcome these limitations by building our approach upon Affinity Propagation Clustering (APC), an exemplar-based clustering approach by Frey [44]. APC has been applied to a variety of problems [34], [35], [53], [88] and extended in multiple ways. [55] uses hard cannot-link constraints between two data points which should not be in the same cluster. Our repellence is much weaker and hence more flexible: it penalizes only when two data points are simultaneously cluster centers, resulting in an significantly different formulation than [55].

3.3 A MESSAGE-PASSING APPROACH FOR NMS

We start in Sec. 3.3.1 by presenting Affinity Propagation Clustering (APC) [44] using its binary formulation [54], which is the most convenient for our extensions. In Sec. 3.3.2, we discuss how we have adapted APC for NMS with a novel inter-cluster repellence term and a relaxation of clustering to remove false positives. We show how the messages must be updated to account for these extensions.

Finally, in Sec. 3.3.3, we propose to use a Latent Structured SVM (LSSVM) [177] to learn the weights of APC.

3.3.1 *Affinity propagation: binary formulation and inference*

Let N be the number of data points and $s(i,j)$ the similarity between data points i and $j \in \{1,\dots,N\}$. APC is a clustering method that relies on data similarities to identify exemplars such that the sum of similarities between exemplars and cluster members is maximized. That is, $s(i,j)$ indicates how well j would serve as an exemplar for i, usually with $s(i,j) \leq 0$ [44]. Following [54], we use a set of N^2 binary variables c_{ij} to encode the exemplar assignment, with $c_{ij} = 1$ if i is represented by j and 0 otherwise. To obtain a valid clustering, the following constraints must hold: (i) each point belongs to exactly one cluster, or equivalently is represented by a single point: $\forall i : \sum_j c_{ij} = 1$; (ii) when j represents any other point i, then j has to represent itself: $\exists i \neq j : c_{ij} = 1 \Rightarrow c_{jj} = 1$. These constraints can be included directly in the objective function of APC:

$$E_{APC}(\{c_{ij}\}) = \sum_{i,j} S_{ij}(c_{ij}) + \sum_i I_i(c_{i1},\dots,c_{iN}) + \sum_j E_j(c_{1j},\dots,c_{Nj}), \quad (3.1)$$

where S_{ij}, I_i and E_j have the following definitions:

$$S_{ij}(c_{ij}) = \begin{cases} s(i,j) & \text{if } c_{ij} = 1 \\ 0 & \text{otherwise,} \end{cases} \quad (3.2)$$

$$I_i(c_{i1},\dots,c_{iN}) = \begin{cases} -\infty & \text{if } \sum_j c_{ij} \neq 1 \\ 0 & \text{otherwise,} \end{cases} \quad (3.3)$$

$$E_j(c_{1j},\dots,c_{Nj}) = \begin{cases} -\infty & \text{if } c_{jj} = 0 \text{ and } \exists i \neq j \text{ s.t. } c_{ij} = 1 \\ 0 & \text{otherwise.} \end{cases} \quad (3.4)$$

Here I_i enforces (i) while E_j enforces (ii). The *self-similarity* $s(i,i)$ favors certain points to be chosen as an exemplar: the stronger $s(i,i)$, the more contribution it makes to eq. (3.1).

The inference of eq. (3.1) is performed by the max-sum message-passing algorithm [44], [54], using two messages: the *availability* α_{ij} (sent from j to i) reflects the accumulated evidence for point i to

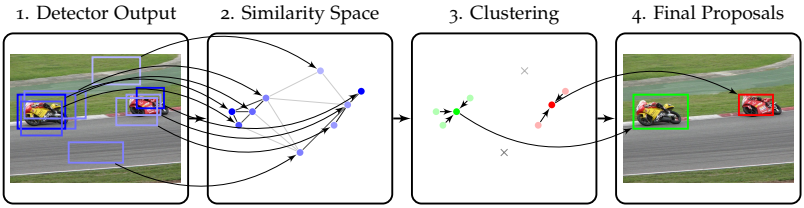


Figure 3.2: Illustration of our NMS pipeline. *1. Detector Output*: the detector returns a set of object window hypotheses with scores. *2. Similarity Space*: the windows are mapped into a similarity space expressing how much they overlap. The intensity of the node color denotes how likely a given box is chosen as an exemplar, the edge strength denotes the similarity. *3. Clustering*: APC now selects exemplars to represent window groups, leaving some windows unassigned. *4. Final Proposals*: the algorithm then returns the exemplars as proposals and removes all other hypotheses.

choose point j as its exemplar, and the *responsibility* ρ_{ij} (sent from i to j) describes how suited j would be as an exemplar for i:

$$\alpha_{ij} = \begin{cases} \sum_{k \neq j} \max(\rho_{kj}, 0) & \text{for } i = j \\ \min(0, \rho_{jj} + \sum_{k \notin \{i,j\}} \max(\rho_{kj}, 0)) & \text{for } i \neq j \end{cases} \tag{3.5}$$

$$\rho_{ij} = s(i,j) - \max_{q \neq j}(s(i,q) + \alpha_{iq}). \tag{3.6}$$

3.3.2 *Adapting affinity propagation for NMS*

We use the windows proposed by the object detector as data points for APC. The self-similarity, or preference to be selected as an exemplar, is naturally chosen as a function of the score of the object detector: the stronger the output, the more likely a data point should be selected. The similarity between two windows is based on their *intersection over union* (*IoU*), as $s(i,j) = \frac{|i \cap j|}{|i \cup j|} - 1$. Here the indices refer to the area of the windows. This expresses the degree

of common area they cover in the image compared to the total area covered which is a good indicator of how likely they describe the same object. To perform competitively, in the following subsections we will extend APC to better suit our needs and present the contributions of this chapter. The resulting processing pipeline is depicted in Fig. 3.2.

Identifying false positives.

False positives are object hypotheses that belong in fact to the background. Therefore, they should not be assigned to any cluster or chosen as an exemplar. This forces to relax constraint (i). To avoid obtaining only empty clusters, this relaxation must be compensated by a penalty for not assigning a data point to any cluster. We do this by modifying eq. (3.3):

$$\widetilde{I}_i(c_{i1}, ..., c_{iN}) = \begin{cases} -\infty & \text{if } \sum_j c_{ij} > 1 \\ \lambda & \text{if } \sum_j c_{ij} = 0 \\ 0 & \text{otherwise.} \end{cases} \tag{3.7}$$

Note how this updated term in eq. (3.1) is equivalent to adding an extra *background* data point that has similarity λ to all the other data points and 0 self-similarity. In the following, the term $\widetilde{I}_i$ will be weighted, hence we can set $\lambda = -1$ without loss of generality.

Inter-Cluster repellence.

In generic object detection the detector precision is much lower compared to detectors trained for a specific object class. To still achieve a high recall it is beneficial to propose a diverse set of windows that covers a larger fraction of the image. However by default, APC does not explicitly penalize choosing exemplars that are very close to each other, as long as they represent their respective clusters well. To encourage diversity among the windows, we therefore propose to include such a penalty by adding an extra term to eq. (3.1).

While this term will favor not selecting windows in the same neighborhood, it will not preclude it strictly either. This will still allow APC to select multiple objects in close vicinity. We denote by

$R = \sum_{i \neq j} R_{ij}(c_{ii}, c_{jj})$ the new set of *repelling* local functions, where, for $i \neq j$:

$$R_{ij}(c_{ii}, c_{jj}) = \begin{cases} r(i,j) & \text{if } c_{ii} = c_{jj} = 1 \\ 0 & \text{otherwise.} \end{cases} \tag{3.8}$$

In other words, we have added a new term for every pair of data points which is active only if both points are exemplars. We penalize this pair by the amount of $r(i,j)$, a *repellence cost*. Again, we base the *repellence cost* between two windows on their *intersection over union*, as $r(i,j) = -\frac{|i \cap j|}{|i \cup j|}$. Note that R_{ij} and R_{ji} refer to the same local function. However we keep both notations for simplicity.

Weights and message passing.

Linearly combining all the above local functions gives us the following new objective function for APC:

$$\widetilde{E}_{APC} = w_a \sum_i S_{ii} + w_b \sum_{i \neq j} S_{ij} + w_c \sum_i \widetilde{I}_i + w_d \sum_{i<j} R_{ij} + \sum_j E_j. \tag{3.9}$$

We have omitted the c_{ij} variables for the sake of clarity, and we have further separated data similarities and self-similarities. Note that the local functions are defined so that all weights are expected to be positive.

Weights are only added to the 4 finite terms and only their relative weight matters for inference. Similar to the original APC, we perform inference, *i.e.*, find the values of $\{c_{ij}\}$ that maximize eq. (3.9) using message-passing. In short, the new terms in eq. (3.9), especially the repellence ones, lead to new messages to be passed between windows. For the sake of space, we show the factor graph corresponding to eq. (3.9) and the full derivation of the 6 corresponding messages in the appendix in Chapter 8. We illustrate them in Fig. 3.3.

The 6 messages (α, β, ρ, η, γ and ϕ) are reduced to 4 (α, ρ, γ and ϕ) by using substitution and integrating the weights back into the local functions. We view the *background* data point as the N+1-th entry in the similarity matrix and can thereby further simplify the

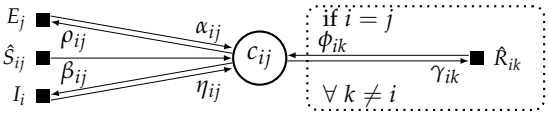


Figure 3.3: The 6 messages passed between variables in our extension of Affinity Propagation are α, β, ρ, η, γ and ϕ.

derivation for the message passing. Then we have 2 messages for all variables c_{ij}:

$$\rho_{ij} = \begin{cases} \hat{s}(i,i) - \max\limits_{q \neq i}(\hat{s}(i,q) + \alpha_{iq}) + \sum_{l \neq i} \phi_{il} & \text{for } i = j \\ \hat{s}(i,j) - \max(\max\limits_{q \notin \{i,j\}}(\hat{s}(i,q) + \alpha_{iq}), \hat{s}(i,i) + \alpha_{ii} + \sum_{l \neq i} \phi_{il}) & \text{for } i \neq j, \end{cases} \tag{3.10}$$

$$\alpha_{ij} = \begin{cases} \sum_{k \neq j} \max(\rho_{kj}, 0) & \text{for } i = j \\ \min(0, \rho_{jj} + \sum_{k \notin \{i,j\}} \max(\rho_{kj}, 0)) & \text{for } i \neq j. \end{cases} \tag{3.11}$$

Additionally, we have 2 messages essentially resulting from the new R_{ij} term which only exist between the subset $\{c_{ii}\}$ of variables:

$$\gamma_{ik} = \hat{s}(i,i) + \alpha_{ii} - \max_{q \neq i}(\hat{s}(i,q) + \alpha_{iq}) + \sum_{l \notin \{i,k\}} \phi_{il} \tag{3.12}$$

$$\phi_{ik} = \max(0, \gamma_{ki} + \hat{r}(i,k)) - \max(0, \gamma_{ki}). \tag{3.13}$$

Following the original message-passing algorithm for APC [44], [54], we initialize all messages with 0. We then iteratively update the messages until convergence.

3.3.3 *Structured learning for affinity propagation*

We address now the problem of learning the weights w_a, w_b, w_c and w_d of eq. (3.9) from training data so as to maximize the performance of the NMS procedure. The training data consists of images with N object window hypotheses and K ground-truth bounding-box annotations for the corresponding object category. The best possible output $\{c^*_{ij}\}$ of APC for those ground-truth bounding-boxes is to keep the proposal with the highest overlap for each

ground-truth bounding-box as long as its *IoU* is at least 0.5. All other proposal should be discarded. This directly determines the target values c^*_{ii} of all c_{ii}. However, correctly setting target values for the remaining c_{ij} $(i \neq j)$ is not straightforward, as we cannot automatically decide which object was detected by this imprecise localization, or whether this window is better modeled as a false positive. Hence, we treat c_{ij} for $i \neq j$ as latent variables. This splits the set of variables in two subsets for each image n: $y_n = \{c^n_{11}, c^n_{22}, ..., c^n_{NN}\}$ are the *observed* variables, with their target y^*_n, and
$z_n = \{c^n_{12}, ..., c^n_{1N}, c^n_{21}, c^n_{23}, ..., c^n_{N-1,N}\}$ the *latent* ones.

We can now rewrite our objective function for image n as: $\widetilde{E}^n_{APC}(y_n, z_n; \vec{w}) = \vec{w}^\top \Psi_n(y_n, z_n)$, where Ψ_n is the concatenation of the terms in eq. (3.9) in a vector, and $\vec{w} = [w_a, w_b, w_c, w_d, 1]^\top$. To learn $\vec{w}$, we resort to Structured-output SVM with latent variables (LSSVM) [177]. This consists of the following optimization problem:

$$\begin{aligned} &\operatorname{argmin}_{\vec{w}\in\mathbb{R}^D, \xi\in\mathbb{R}^n_+} \ \frac{\lambda}{2}||\vec{w}||^2 + \sum_n \xi^n \\ &\text{s.t. } \forall n, \max_{\hat{z}_n} \widetilde{E}^n_{APC}(y^*_n, \hat{z}_n; \vec{w}) \geq \max_{y_n, z_n} \left(\widetilde{E}^n_{APC}(y_n, z_n; \vec{w}) + \Delta(y_n, y^*_n) \right) - \xi^n, \end{aligned} \tag{3.14}$$

where ξ^n are slack variables, and Δ is a loss measuring how y_n differs from y^*_n. This is equivalent to finding a $\vec{w}$ which maximizes the energy of APC for the target variables y^*_n, by a margin Δ, independent of the assignment of z_n. Following [177], we solve eq. (3.14) using the concave-convex procedure (CCCP) [178] and the Structured-output SVM implementation by [153]. We define Δ:

$$\Delta(y, y^*) = \sum_i \nu[c_{ii} - c^*_{ii} < 0] + \pi \left(1 - \max_{\text{obj}} \frac{|i \cap obj|}{|i \cup obj|} \right) [c_{ii} - c^*_{ii} > 0]. \tag{3.15}$$

where $\nu \geq 0$ is the cost for not choosing a window as an exemplar although it is the best candidate for one of the objects. When a box is chosen as an exemplar even though it is not the best candidate it is considered as a false positive. This is smoothly penalized by $\pi \geq 0$ by considering the overlap with the ground-truth object it most overlaps with. The values for π and ν are chosen depending

on the application, usually $\nu/\pi > 1$. Using CCCP additionally implies that we are able to perform loss-augmented inference (*i.e.*, find (y_n, z_n) that maximizes the right-hand side of the constraints in eq. (3.14)), and partial inference of z_n (*i.e.*, the left-hand side of the constraint). For the left-hand side, $\operatorname{argmax}_{\hat{z}} \widetilde{E}_{APC}(y^*, \hat{z}; \vec{w})$ can be computed directly. Given the cluster centers y_n^* we just assign all other boxes which are not cluster centers to the most similar clusters. For false positives, this could also be the *background* data point depending on the current value for w_c. This results in a valid clustering which maximizes the total similarity for the given exemplars.

Concerning the right-hand side, we can easily incorporate Δ as an extra term in eq. (3.9), and use message passing to obtain the corresponding (y_n, z_n). When incorporating the loss term into the message passing, only the similarity $\hat{s}$ needs to be modified, leading to $\hat{s}_\Delta$:

$$\hat{s}_\Delta(i,j) = \begin{cases} \hat{s}(i,j) - \nu & \text{for } i=j \text{ and } c_{ii}^n = 1 \\ \hat{s}(i,j) + \pi \left(1 - \max_{\text{obj}} \frac{|i \cap obj|}{|i \cup obj|}\right) & \text{for } i=j \text{ and } c_{ii}^n = 0 \\ \hat{s}(i,j) & \text{otherwise.} \end{cases} \tag{3.16}$$

3.4 EXPERIMENTS ON OBJECT CLASS DETECTION

To compare the proposed exemplar based clustering framework to Greedy NMS, we measured their respective performance for object class detection. We are especially interested in the cases we presented in Fig. 3.1 where Greedy NMS fails, and we will present insights why our proposed method handles these better. A detailed analysis will address localization errors (Fig. 3.1 (b)), closeby labeled objects (Fig. 3.1 (a)), precision as well as detections on background (Fig. 3.1 (c)). This is in line with Hoiem's [68] in-depth analysis of the performance of a detector, not only giving a better understanding of its weaknesses and strengths but also showing that specific improvements are necessary to advance in object detection.

3.4.1 *Implementation details*

In this section the clustering is applied to Felzenszwalb's [42] (release 5) object class detector based on a deformable parts model (DPM). Performance is measured on the widely used Pascal VOC 2007 [39] dataset composed of 9,963 images containing objects of 20 different classes. We keep the split between training and testing data as described in [42]. The DPM training parameters are set to their default values. We keep all windows with a score above a threshold which is determined for each class during training but at most 250 per image. The similarity between two windows is based on their *intersection over union*, as described in Sec. 3.3. As the score of the Felzenszwalb boxes p is not fixed to a range, it is scaled to $[-1, 0]$ by a sigmoidal function $s(i, i) = \frac{1}{1+e^{-p}} - 1$. The presented results for *APC* are trained following Sec. 3.3.3 on the validation set. For a fair comparison, the ratio ν/π was set to yield a total number of windows similar to Greedy NMS.

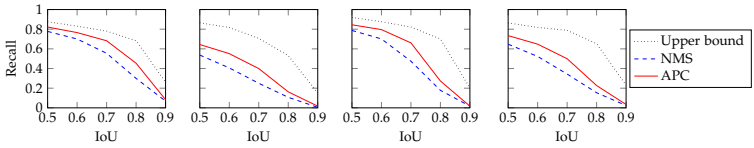
3.4.2 *Results*

The results are presented in separate subsections that compare the performance of APC and Greedy NMS with emphasis on the specific issues presented in Fig. 3.1.

Can APC provide better-fitting boxes than Greedy NMS (Fig. 3.1 (b))?

Here we show that solving NMS globally through clustering can help to select better-fitting bounding-boxes compared to Greedy NMS. We look at the detection rate for different *IoU* thresholds with the object for detection. The upper bound is determined by the detection rate of the detector when returning all windows, *i.e.* without any NMS.

The quantitative results in Fig. 3.4 confirm that APC recovers more objects with the same number of boxes compared to Greedy NMS, especially performing well when a more precise location of the object is required ($IoU \geq 0.7$). We then evaluated the area under the curve in Fig. 3.4 for each class separately (normalized to 1), whose values are shown in Tab. 3.1. Here we perform better



(a) bicycle (b) chair (c) horse (d) average

Figure 3.4: Object class detection: *IoU* vs. recall for a selection of classes (a-c) as well as the average across all (d). Our method consistently outperforms Greedy NMS for different *IoU* thresholds.

Table 3.1: Object class detection: area under curve (AUC) for *IoU* vs. recall.

	aeroplane	bicycle	bird	boat	bottle	bus	car	cat	chair	cow	diningtable
Upper bound	0.592	0.716	0.495	0.476	0.482	0.744	0.663	0.718	0.641	0.600	0.788
NMS	0.303	0.494	0.170	0.187	0.288	0.450	0.432	0.335	0.259	0.312	0.391
APC	**0.426**	**0.589**	**0.297**	**0.260**	**0.333**	**0.552**	**0.498**	**0.432**	**0.361**	**0.426**	**0.556**

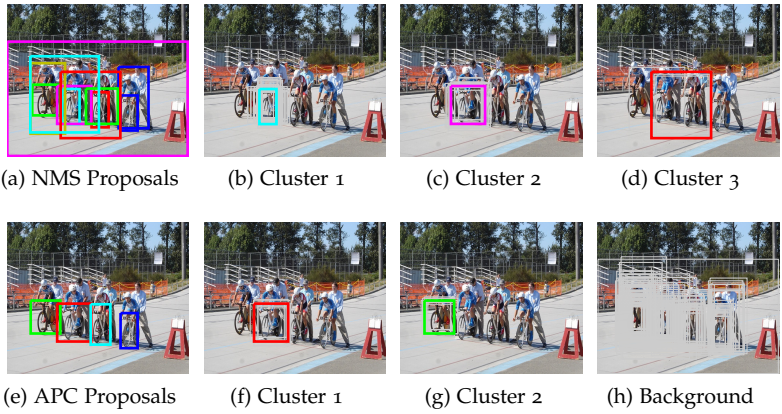
	dog	horse	motorbike	person	pottedplant	sheep	sofa	train	tvmonitor	**average**
Upper bound	0.685	0.740	0.727	0.620	0.508	0.497	0.855	0.707	0.702	0.648
NMS	0.265	0.439	0.422	0.320	0.170	0.200	0.470	0.394	0.482	0.339
APC	**0.336**	**0.540**	**0.522**	**0.418**	**0.303**	**0.322**	**0.584**	**0.533**	**0.510**	**0.440**

across all classes with an increase between 0.17 for the *diningtable* class and 0.03 for the *tvmonitor* class. On average the AUC can be increased from 0.34 to 0.44. Even though selecting the right boxes from the output of the detector could have led up to an AUC of 0.65, APC was still able to narrow the gap by almost a third.

This is also confirmed by the qualitative results in Fig. 3.5: whereas NMS proposes several boxes for the same bike (e.g. (b), (c)) and even sometimes proposes one box covering two objects (d), our method returns one box per bike ((f), (g)). These boxes are the exemplars of clusters only containing boxes which tightly fit the bikes – the others are collected in the background cluster (h).

Does APC avoid to suppress objects in groups (Fig. 3.1 (a))?

Two (or more) objects form a group if they at least touch each other ($IoU > 0$). Thus we remove from the ground-truth the objects that do not overlap with any other object of the same class, and compute the recall (with $IoU = 0.5$) on the remaining objects for



(a) NMS Proposals (b) Cluster 1 (c) Cluster 2 (d) Cluster 3

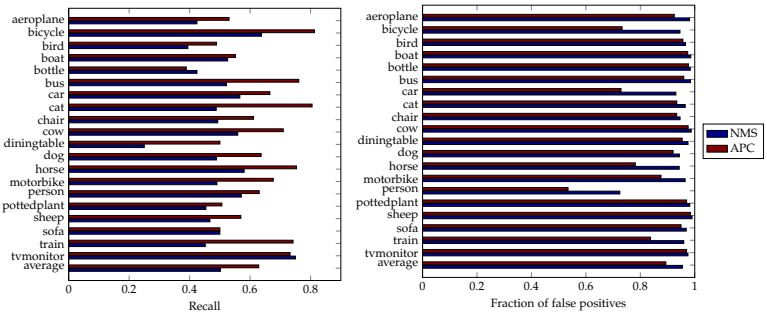
(e) APC Proposals (f) Cluster 1 (g) Cluster 2 (h) Background

Figure 3.5: Object class detection: qualitative results. These figures show an example of the proposed windows. The colored box are the exemplars for the gray boxes. Upper row: Greedy NMS. Lower row: APC.

the same number of proposed windows as shown in Fig. 3.6a. On average APC recovers 62.9% objects vs. 50.2% for Greedy NMS, with an increase of up to 31.7% for individual classes. Noting that these objects are especially difficult to detect, APC is more robust at handling nearby detector responses. This is a clear advantage of the proposed clustering based approach.

Can APC suppress more false-positives (Fig. 3.1 (c))?

Already the qualitative results in Fig. 3.5h suggest that the clustering relaxation proposed in Sec. 3.3.2 helps to remove extraneous boxes with low scores which do not describe any object. For a quantitative analysis, we look again at the results of APC and Greedy NMS when both return the same number of windows. Noting that both post-processing algorithms are provided with exactly the same windows by the detector as input, we now evaluate which method is better at suppressing false positives. In this context we define false positives as all boxes which do not touch any object ($IoU = 0$). These boxes are nowhere near detecting an object as



(a) Performance on touching objects (b) Suppressing false positives

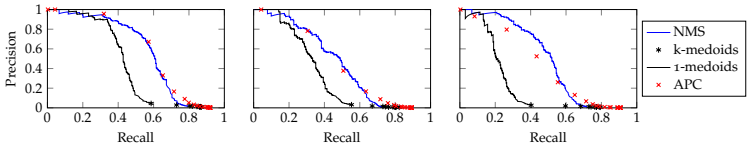
Figure 3.6: Object class detection: in-depth analysis. (a) compares the recall of Greedy NMS and APC on pairs of objects ($IoU > 0$ between objects) – APC recovers significantly more of these rather difficult objects. (b) shows the fraction of false positives – windows that do not touch any object: APC on average reduces the fraction of false positives, with a significant reduction for some classes, *i.e. bicycle, car, person*.

usually at least $IoU \geq 0.5$ is required for detection. As shown in Fig. 3.6b APC is able to reduce the fraction of false positives proposed from 95.5% for NMS to 89.4% with consistent improvement across all classes. For some classes like *bicycle, car* and *person* whose objects often occur next to each other, APC shows significant false positive reduction of up to 21.6%, proposing more relevant windows which also reflects in the recall in Fig. 3.4.

Table 3.2: Object class detection: average precision NMS vs. APC

		aeroplane	bicycle	bird	boat	bottle	bus	car	cat	chair	cow	diningtable
IoU 0.5	NMS	**0.332**	**0.593**	0.103	**0.157**	**0.266**	**0.520**	**0.537**	**0.225**	**0.202**	**0.243**	**0.269**
	APC	0.298	0.511	**0.108**	0.107	0.130	0.369	0.428	0.197	0.149	0.168	0.235
IoU 0.8	NMS	**0.101**	0.198	**0.091**	0.023	**0.096**	**0.135**	**0.123**	0.021	0.057	0.048	0.036
	APC	0.090	**0.222**	**0.091**	**0.091**	0.092	0.114	0.112	**0.093**	**0.093**	**0.092**	**0.100**

		dog	horse	motorbike	person	pottedplant	sheep	sofa	train	tvmonitor	**mAP**	**"mAP"**
IoU 0.5	NMS	0.126	0.565	**0.485**	**0.433**	**0.135**	**0.209**	**0.359**	**0.452**	**0.421**	**0.332**	
	APC	**0.129**	**0.579**	0.432	0.363	0.116	0.143	0.259	0.449	0.175		0.267
IoU 0.8	NMS	0.004	0.061	0.126	0.106	0.006	0.030	0.105	0.044	**0.144**	0.078	
	APC	**0.091**	**0.122**	**0.128**	**0.111**	**0.091**	**0.091**	**0.115**	**0.104**	0.107		**0.108**

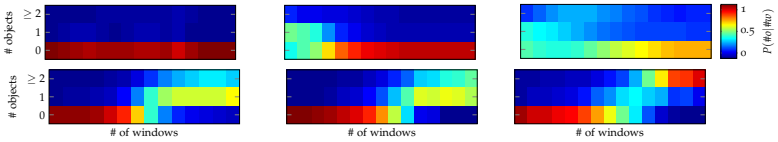


(a) horse (b) train (c) motorbike

Figure 3.7: Object class detection: precision vs. recall. The precision-recall curves reveal that APC performs competitively compared to Greedy NMS at a similar precision but higher recall while significantly outperforming k-medoids.

What is the precision of APC compared to NMS and k-medoids?

We now vary the ratio of the training parameters ν / π. APC returns a fixed set of boxes, ranging from less than a box up to several hundreds per image depending on the clustering parameters which are obtained through training by setting this ratio for the specific application. These boxes, although they cover the objects well, do not follow any kind of ranking as they altogether form the result of a globally solved problem. Since AP is designed to measure the performance of a ranking system, it is simply not appropriate for APC, as that would require that one can select the best possible subset of the proposed boxes. Still, we computed a proxy to AP by linearly interpolating the precision for points of consecutive recall (which need not be consecutive values of the varied parameter). This results in a "mAP" for APC of 0.27 compared to a real mAP of 0.33 for greedy NMS as shown in Tab. 3.2. AP is mostly influenced by the highest scored detections, so greedy NMS at an IoU of 0.5 is hard to beat with the same underlying detector. However, as such, AP does not reward methods with more precise object localizations than 0.5 and overall better recall. These are precisely areas where greedy NMS can be improved, and therefore we resorted to a deeper analysis. As a matter of fact, if we set a more difficult detection criterion of, e.g., 0.8 IoU, then APC outperforms greedy NMS with a "mAP" of 0.11 compared to 0.08. This is another aspect where APC shows superior performance compared to



(a) bicycle **(b)** car **(c)** person

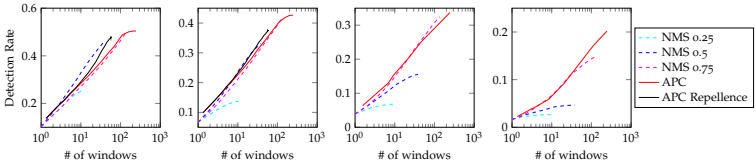
Figure 3.8: Object class detection: predicting the number of objects. Greedy NMS approximately returns the same number of boxes independent of the number of objects in the image. Therefore the posterior $P(\#\,\text{objects}\,|\,\#\,\text{windows})$ remains uninformative about the object count. In contrast, APC is very flexible and adjust the number of windows being returned depending on how many objects there are in the image.

greedy NMS. As each clustering has a well-defined precision and recall, we can have a scatter plot to compare it to Greedy NMS. Fig. 3.7 shows that APC achieves a similar precision at low recall but better recall at low precision.

We also compared APC to a k-medoids clustering baseline using the same similarity as for APC. To account for the score of the proposals, the self-similarity of the k selected cluster centers (varied from 1 to 10) was added to the overall cost function to favor boxes with better scores. k-medoids leads to similar precision-recall scatter plots as shown in Fig. 3.7. Additionally, we plot the precision-recall curve for $k = 1$ (*1-medoids*) by ranking the cluster centers with their original scores. As shown in Fig. 3.7 already in the case of *1-medoids* many objects are recovered. However, the precision drops for larger recalls since it predicts k objects in every single image. This lack of flexibility is a clear disadvantage of k-medoids and other similar clustering algorithms compared to APC.

Does APC better predict the number of objects in the image?

Studying the experimental results revealed that Greedy NMS approximately returns the same number of boxes per image independent of whether there was an object in the image. In contrast, for



(a) 0.5 *IoU* (b) 0.6 *IoU* (c) 0.7 *IoU* (d) 0.8 *IoU*

Figure 3.9: Generic object detection: Greedy NMS requires to adopt the parameter for suppression for different *IoU* thresholds to always perform competitively. In contrast, APC performs consistently well, beating Greedy NMS especially for precise object detection ($IoU \geq 0.7$). Introducing a repellence helps to boost performance for less precise object detection by enforcing diversity among the proposed windows.

APC it greatly varied between images. Therefore, we simply measured the posterior probability
$P(\#\text{ objects} \mid \#\text{ windows})$. Fig. 3.8 depicts this probability for both Greedy NMS and APC for a selection of classes. For Greedy NMS (upper row in Fig. 3.8) the number of proposed windows is mostly uninformative regarding how many objects there are in the image. In comparison for APC (lower row in Fig. 3.8), there is a strong correlation between the number of windows proposed and the likelihood that there are 1 or more objects: given the number of windows APC proposes we can estimate how many objects there are in the image.

3.5 EXPERIMENTS ON GENERIC OBJECT DETECTION

We apply APC to generic object detection which gained popularity in recent years as a preprocessing step for many state-of-the-art object detectors [138], [150]. We use the objectness measure introduced by [3] which is the only one to provide a probability p with the window it proposes, unlike [101], [119], [150].

3.5.1 *Implementation details*

Performance is again evaluated on Pascal VOC 2007 where we split the dataset in the same way as in [2] and used the classes *bird, car, cat, cow, dog, sheep* for training the objectness algorithm as well as the clustering and the remaining 14 classes for testing. Images which had occurrences of both training and testing classes were dropped and in contrast to [2] we also kept objects marked as difficult and truncated. The self-similarity is based on the probability of containing an object $s(i,i) = p(i) - 1$ and the similarity between boxes is defined by the overlap. We sampled 250 windows with multinomial sampling which still allows to recover a large fraction of the objects. As presented in [2], Greedy NMS significantly improved the detection rate for objectness. This motivates our experiments where we compare Greedy NMS against APC.

3.5.2 *Results*

After training APC, we compare its detection rate with Greedy NMS for different *IoU* thresholds with the object. For APC we show the performance both without and with repellence; for NMS we varied the threshold for suppression. Looking at Fig. 3.9, we make 3 observations: (i) when proposing very few windows per image (< 10) APC typically performs better than Greedy NMS. (ii) for an $IoU \geq 0.7$ the standard NMS threshold of 0.5 performs significantly worse than APC. This requires that Greedy NMS reruns with a higher threshold for suppression. In comparison our method is much more consistent across varying *IoU*. (iii) for APC diversity can be enforced by activating the inter-cluster repellence which avoids having cluster centers close-by each other. This boosts our performance for $IoU \leq 0.6$ by close to up to 5% from 42.9% to 47.5% for $IoU = 0.5$.

3.6 DISCUSSION

We presented a novel clustering-based NMS algorithm based on Affinity Propagation. We showed that it successfully tackles short-

comings of Greedy NMS for object class and generic object detection.

Specifically we show that our method – whose parameters can be learned automatically depending on the application – yields better-fitting bounding-boxes, reduces false positives, handles close-by objects better and is better able to predict the number of objects in an image, all at a competitive precision compared to Greedy NMS. Given that APC tries to find a global solution to the NMS problem, it is however computationally more complex and still relatively slow, taking approximately $1s$ to cluster 250 bounding-boxes. In the future, we therefore plan to explore approximative solutions.

APC could also be expanded to multi-class object detection, integrating context and holistic knowledge. The newly-introduced repellence could be based not only on the overlap between the boxes, but rather the similarity in appearance expressing how likely the two windows cover the same object. In future work, we want to learn the window similarity by considering visual-features that may help to distinguish between multiple detections of the same object or nearby objects. We are convinced that APC can be of interest for many other areas where NMS is used, *e.g.* edge detection [14], [32].

In the next two chapters, we present fine-grained classification methods for age, gender and attractiveness of faces. All those method require first the detection of the face in the image and thus the presented method in this chapter can be seen as a preprocessing step.

4 PREDICTING REAL AND APPARENT AGE

4.1 INTRODUCTION

Following the detection of objects in the previous chapter, this chapter aims at age estimation from a single face image (see Fig. 4.1) which is an important task in human and computer vision and has many applications such as in forensics or social media. It is closely related to the prediction of other biometrics and facial attributes tasks, such as gender, ethnicity, hair color and expressions. A large amount of research has been devoted to age estimation from a face image under its most known form – the real, biological, age estimation. This research spans decades as summarized in large studies [17], [36], [57], [62], [111]. Several public standard datasets [17], [111], [118] for real age estimation permit public performance comparison of the proposed methods. In contrast, the study of apparent age, that is the age as perceived by other humans, is in its early stages. The ChaLearn Looking At People ICCV 2015 challenge [38] provided the largest dataset known to date of images with apparent age annotations, here called the LAP dataset, and 115 registered teams proposed novel solutions to the problem.

Figure 4.1: Predicting the real and apparent age of a person.

With the recent rapid emergence of the intelligent applications there is a growing demand for automatic extraction of biometric information from face images or videos. Applications where age estimation can play an important role include: (i) access control, e.g., restricting the access of minors to sensible products like alcohol from vending machines or to events with adult content; (ii) human-computer interaction (HCI), e.g., by a smart agent estimating the age of a nearby person or an advertisement board adapting its offer for young, adult, or elderly people, accordingly; (iii) law enforcement, e.g., automatic scanning of video records for suspects with an age estimation can help during investigations; (iv) surveillance, e.g., automatic detection of unattended children at unusual hours and places; (v) perceived age, e.g., there is a large interest of the general public in the perceived age (c.f. howhot.io), also relevant when assessing plastic surgery, facial beauty product development, theater and movie role casting, or human resources help for public age specific role employment.

One should note that the intelligent applications need to tackle age estimation under unconstrained settings, that is, the face is not aligned, and under known, unchanged, light and background conditions. Therefore, in the wild, a face needs first to be detected, then aligned, and, finally, used as input for an age estimator. It is particularly this setup we target in this chapter with our system. Despite the recent progress [38], [111], [128] the handling of faces in the wild and the accurate prediction of age remains a challenging problem.

4.1.1 *Proposed method*

Our approach – called Deep EXpectation (DEX) – to age estimation is motivated by the recent advances in fields such as image classification [23], [77], [129] or object detection [52] fueled by deep learning. From the deep learning literature we learn four key ideas that we apply to our solution: (i) the deeper the neural networks (by sheer increase of parameters / model complexity) are the better the capacity to model highly non-linear transformations - with some optimal depth on current architectures as [65] suggests; (ii) the larger and more diverse the datasets used for training are the

better the network learns to generalize and the more robust it becomes to over-fitting; (iii) the alignment of the object in the input image impacts the overall performance; (iv) when the training data is small the best is to fine-tune a network pre-trained for comparable inputs and goals and thus to benefit from the transferred knowledge.

We always start by first rotating the input image at different angles to then pick the face detection [103] with the highest score. We align the face using the angle and crop it for the subsequent steps. This is a simple but robust procedure which does not involve facial landmark detection. For our convolutional neural networks (CNNs) we use the deep VGG-16 architecture [134]. We always start from pre-trained CNNs on the large ImageNet [129] dataset for image classification such that (i) to benefit from the representation learned to discriminate 1000 object categories in images, and (ii) to have a meaningful representation and a warm start for further re-training or fine-tuning on relatively small(er) face datasets. Fine-tuning the CNNs on face images with age annotations is a necessary step for superior performance, as the CNN adapts to best fit to the particular data distribution and target of age estimation. Due to the scarcity of face images with (apparent) age annotation, we explore the benefit of fine-tuning over crawled Internet face images with available (biological, real) age. We crawl 523,051 face images from the IMDb and Wikipedia websites to form IMDB-WIKI - our new dataset which we make publicly available. Fig. 4.4 shows some images. It is the largest public dataset with gender and real age annotations. While age estimation is a regression problem, we go further and cast the age estimation as a multi-class classification of age bins followed by a softmax expected value refinement.

Our main contributions are as follows:

1. the IMDB-WIKI dataset, the largest dataset with real age and gender annotations;

2. a novel regression formulation through a deep classification followed by expected value refinement;

3. the DEX system, winner of the LAP 2015 challenge [38] on apparent age estimation.

This chapter is an extended and detailed version of our previous LAP challenge report paper [125]. We now officially introduce our IMDB-WIKI dataset for apparent age estimation, provide a more in depth analysis of the proposed DEX system, and apply the method and report results also on standard real age estimation datasets.

The remainder of the chapter is organized as follows. Section 4.2 briefly reviews related age estimation literature. Section 4.3 introduces our proposed method (DEX). Section 4.4 introduces publicly our new IMDB-WIKI dataset with faces in the wild and age and gender labels, then describes the experimental setups and discusses the achieved results. Section 4.5 concludes the chapter.

4.2 RELATED WORK

While almost all literature prior the LAP 2015 challenge focuses on real (biological) age estimation from a face image, Han *et al*. [63] provide a study on demographic estimation in relation to human perception and machine performance. In the next, we briefly review the age estimation literature and describe a couple of methods that most relate with our proposed method. We refer to [17], [36], [46], [57], [63], [111] for broader literature reviews.

4.2.1 *Real age estimation*

Most of the prior literature assumes a normalized (frontal) view of the face in the input image or employ a face preprocessing step such that the face is localized and an alignment of the face is determined for the subsequent processing steps. Generally, the age estimators work on a number of extracted features, feature representations and learn models from training data such that to minimize the age estimation error on a validation data. The whole process assumes that the train, validation, and test data have the same distribution and are captured under the same conditions.

FG-NET [111] and MORPH [118] datasets with face images and (real) age labels are the most used datasets allowing for comparison of methods and performance reporting under the same benchmarking conditions. We refer to [111] for an overview of research

(365+ indexed papers) on facial aging with results reported on FG-NET dataset.

A large number of face models has been proposed. We follow the taxonomy from [57] and mention: wrinkle models [81], anthropometric models [40], [81], [114], active appearance models (AAM) [24], aging pattern subspace [51], age manifold [47], [58], [61], biologically-inspired models (including biologically-inspired features (BIF) [105]), compositional and dynamic models [136], [166], local spatially flexible patches [169], and methods using fast Fourier transform (FFT) and genetic algorithm (GA) for feature extraction and selection [48], local binary patterns (LBP) [173], Gabor filters [49]. Recently, the convolutional neural networks (CNN) [90], biologically inspired, were successfully deployed for face modeling and age estimation [92], [151], [159].

The age estimation problem can be seen as a regression [47] or as a classification problem up to a quantization error [51], [84]. Among the most popular regression techniques we mention Support Vector Regression (SVR) [33], Partial Least Squares (PLS) [50], Canonical Correlation Analysis (CCA) [64], while for classification the traditional nearest neighbor (NN) and Support Vector Machines (SVMs) [25].

In the next we select a couple of the representative (real) age estimation methods. Yan *et al*. [168] employ a regressor learning from uncertain labels, Guo *et al*. [58] learn a manifold and local SVRs, Han *et al*. [63] apply age group classification and within group regression (DIF), Geng *et al*. [51] introduce AGES (AGing pattErn Subspace), Zhang *et al*. [182] propose a multi-task warped gaussian process (MTWGP), Chen *et al*. [20] derive CA-SVR with a cumulative attribute space and SVR, Chang *et al*. [16] rank hyperplanes for age estimation (OHRank), Huerta *et al*. [70] fuse texture and local appearance descriptors, Luu *et al*. [97] use AAM and SVR, while Guo and Mu [59] use CCA and PLS.

Recently, Yi *et al*. [175] deployed a multi-scale CNN, Wang *et al*. [159] used deep learned features (DLA) in a CNN way, while Rothe *et al*. [128] went deeper with CNNs and SVR for accurate real age estimation on top of the CNN learned features.

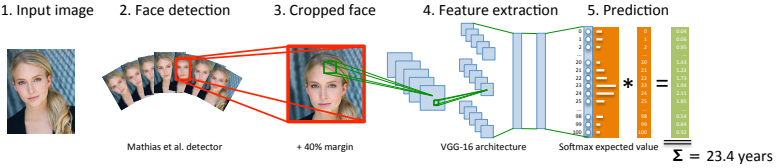


Figure 4.2: Pipeline of DEX method for age estimation.

4.2.2 *Apparent age estimation*

Our DEX [125] method (CVL_ETHZ team, 1st place in LAP challenge) was initially introduced for apparent age estimation at the ChaLearn LAP 2015 challenge [38]. This chapter is mostly based on [124] and an extension of [125], releasing the IMDB-WIKI age estimation dataset with some in-depth analysis. Furthermore, this chapter shows that the model presented in [125] achieves state-of-the-art also on real age estimation. Some more detailed qualitative and quantitative evaluations in this chapter confirm the robustness and good performance of the DEX model. We review several runner-up methods that relate the most to our work and refer to [38] and Section 4.4.2 for more details on the LAP challenge. These methods are representative since LAP is the largest dataset to date on apparent age estimation and the methods employ deep learning and are the best out of 115 registered participants. A note is due: all the following apparent age estimation techniques are either pre-trained for real age estimation or can easily be adapted to it.

Liu *et al*. [96] (ICT-VIPL team, 2nd place in LAP challenge [38], see Tab. 4.3) proposes the following approach based on general to specific deep transfer learning and GoogleNet architecture [137] for 22-layers CNN. 1) Pre-train CNN for multi-class face classification using the CASIA-WebFace database and softmax loss; 2) Fine-tune CNN for age estimation on large extra age dataset with two losses: Euclidean for age encoding and cross-entropy loss of label distribution learning based age encoding; 3) Fine-tune CNN on the LAP apparent age data; 4) Ensemble Learning and fusion of 10 CNNs.

Zhu *et al*. [184] (WVU_CVL team, 3rd place in LAP challenge) employ GoogleNet [137] deep CNN networks trained on thousands of public facial images with real age labels. These are then fine-tuned on LAP apparent age data and then the CNN features are extracted. Random Forest and SVR are learned on each of ten age groups for age estimation and then their results are fused at test time.

Yang *et al*. [171] (SEU-NJU team, 4th place in LAP challenge) use face and landmark detection for face alignment and the VGG-16 architecture [137] for modeling. Private and MORPH 2 data are used for training of multiple networks with different setups, aligned and non-aligned faces, different color spaces, filters, objective losses. The final prediction is a fusion.

UMD team (5th place in LAP challenge) employs face and landmark detection [74], a CNN model [19], Adience [36] and MORPH datasets. A classification in three age groups is followed by age regression.

Enjuto team (6th place in LAP challenge) use face detection [103] and face landmark detection [74] and 6 CNNs for classification in three age groups and for local (part face) and global (whole face) prediction of age. The results are fused.

4.3 PROPOSED METHOD (DEX)

The proposed method, DEX (Deep EXpectation) follows the pipeline in Fig. 4.2. In this section each step of the pipeline is explained in detail.

4.3.1 *Face alignment*

As many datasets used in this chapter do not show centered frontal faces but rather faces in the wild (cf. Fig. 4.2 (1) and Fig. 4.4), we detect and align the faces for both training and testing.

An ideal input face image should be of the same or comparable size, centered, and aligned to a normalized position and with minimum background. We choose the off-the-shelf Mathias *et al*. [103] face detector to obtain the location and size (scale) of the face in

each image. This state-of-the-art face detector uses the Deformable Parts Model (DPM) [42] and inherits robust performance. As expected, by cropping the detected face for the following age estimation processing instead of using the entire image we obtain massive increases in performance.

Many approaches employ rather complex alignment procedures involving accurate facial landmark detectors and image warping [139], [171]. In our preliminary experiments we observed that the failure of the landmark detectors is difficult to predict and harms the performance as it leads to wrong face alignments. Since we target faces in the wild, use a robust face detector, and our CNNs can tolerate small alignment errors, we build our alignment procedure as follows.

We explicitly handle rotation by running the detector not only on the original image but on images rotated with steps of 5° (cf. Fig. 4.2 (2)). Due to limited computational resources we check only angles between $-60°$ and 60°. Additionally we run the detector at $-90°$,90° and 180° to cope with flipped or rotated images. At the end the face with the highest detection score across all rotations is picked and then rotated to up-frontal position.

For very few face images, the detector is unable to detect a face (*i.e.* less than 0.2% for the LAP dataset). In those cases the entire image is taken as the face. We notice that performance increases when considering also the context around the face. Therefore we extend the detected face by taking additional 40% of the width and height of the face on all sides (cf. Fig. 4.2 (3)). If the face is too large so that there is no context on some of the sides, the last pixel at the border is just repeated. This ensures that the face is always placed in the same location in the image. As the aspect ratio of the resulting image might differ, it is squeezed to 256×256 pixels. This forms the input for the deep convolutional neural network.

Our above-mentioned face alignment procedure is robust as it involves a robust face detector and searches using brute-force for the rotation angle of the face. A minus is that it provides a rather rough alignment when compared to more involved facial landmark detection methods. However, in our experiments the facial landmark detectors work very well for near-frontal faces while for the difficult cases produce inaccurate results. The failure cases in

facial landmark detection are difficult to predict and lead to failure in alignment, therefore harm the performance. Therefore we refer to our face alignment procedure as "robust", since there are very few cases where it fails completely and gives always a rough alignment. Though our procedure does not provide very precise pixel-wise alignments, our CNN copes well with such level of precision.

4.3.2 *Age estimation*

We employ a convolutional neural network (CNN) to predict the age of a person starting from a single input face image. This takes an aligned face with context as input and returns a prediction for the age. The CNN is trained on face images with known age.

CNN architecture

Our method uses a CNN with the VGG-16 [134] architecture (cf. Fig. 4.2 (4)). Our choice is motivated (i) by the deep but manageable architecture, (ii) by the impressive results achieved using VGG-16 on the ImageNet challenge [129], (iii) by the fact that as in our case the VGG-16 architecture starts from an input image of medium resolution (256×256), (iv) and that pre-trained models for classification are publicly available allowing warm starts for training.

The VGG-16 architecture is much deeper than previous architectures such as the AlexNet [77] with 16 layers in total, 13 convolutional and 3 fully connected layers. It can be characterized by its small convolutional filters of 3x3 pixels with a stride of 1. AlexNet in comparison employs much larger filters with a size of up to 11×11 at a stride of 4. Thereby each filter in VGG-16 captures simpler geometrical structures but in comparison allows more complex reasoning through its increased depth.

For all our experiments we start with the convolutional neural network pre-trained on the ImageNet images, the same models used in [134]. Unless otherwise noted, we fine-tune the CNN on the images from the newly introduced IMDB-WIKI dataset to adapt to face image contents and age estimation. Finally, we tune

the network on the training part of each actual dataset on which we evaluate. The fine-tuning allows the CNN to pick up the particularities, the distribution, and the bias of each dataset and thus to maximize the performance.

Table 4.1: Performance on validation set of ChaLearn LAP 2015 apparent age estimation challenge. Varying number of output neurons (*last layer initialized with weights from IMDB-WIKI pre-training, † fine-tuned on LAP (Expected Value* 101 setup) before training SVR). conv5_3 (100,352 dim) is the last convolutional layer. fc6 (4,096 dim) and fc7 (4,096 dim) are the penultimate and last fully connected layers, respectively.

		w/o IMDB-WIKI pre-training				w/ IMDB-WIKI pre-training			
	Number	Ranges				Ranges			
Learning strategy	output	Uniform		Balanced		Uniform		Balanced	
	neurons	MAE	ϵ-error	MAE	ϵ-error	MAE	ϵ-error	MAE	ϵ-error
SVR on conv5_3		8.472	0.647			4.570	0.411		
SVR on fc6		15.086	0.787			3.690	0.329		
SVR on fc7		12.083	0.720			3.670	0.321		
SVR on conv5_3†		7.150	0.560			4.020	0.356		
SVR on fc6†		9.695	0.663			3.406	0.297		
SVR on fc7†		9.069	0.664			3.323	0.288		
Regression	1	5.586	0.475			3.650	0.310		
Classification	5	6.953	0.563	6.275	0.501	5.944	0.529	4.394	0.369
Classification	10	6.404	0.511	6.352	0.516	4.243	0.388	3.912	0.337
Classification	25	6.474	0.521	6.507	0.516	3.563	0.309	3.676	0.322
Classification	50	6.424	0.510	7.044	0.555	3.463	0.298	3.517	0.306
Classification	101	7.083	0.548			3.640	0.310		
Classification*	101					3.521	0.305		
Expected Value	5	6.306	0.535	5.589	0.464	5.226	0.481	3.955	0.329
Expected Value	10	5.586	0.470	**5.369**	**0.456**	3.553	0.315	3.505	0.296
Expected Value	25	**5.580**	**0.469**	5.522	0.468	3.306	0.289	3.353	0.290
Expected Value	50	5.653	0.473	6.042	0.509	3.349	0.291	**3.318**	**0.289**
Expected Value	101	5.965	0.493			3.444	0.299		
Expected Value*	101					**3.252**	**0.282**		

4.3.3 *Evaluation protocol*

For quantitative evaluation in our experiments we use two different measures.

MAE. For all experiments we report the Mean Absolute Error (MAE) in years. This is the average of the absolute error between

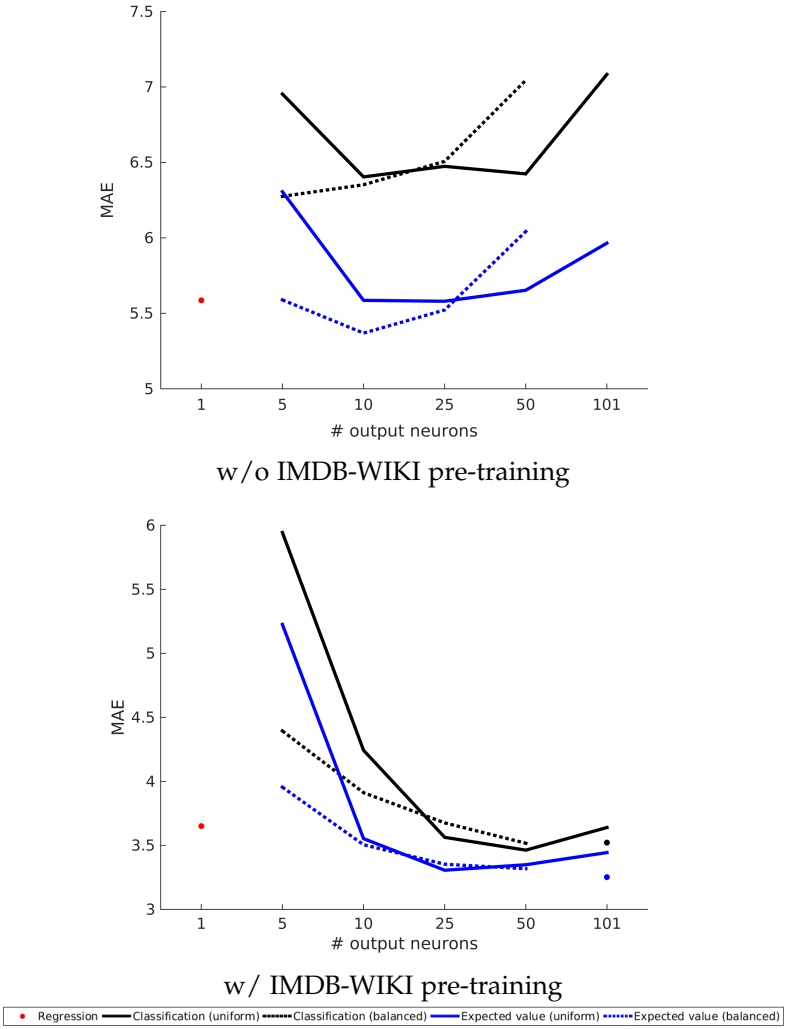


Figure 4.3: Impact of the number of output neurons and the age ranges on the MAE performance.

the predicted age and the ground truth age. MAE is the most used

measure in the literature, a *de facto* standard for age estimation.

ϵ-error. The LAP challenge proposes the ϵ-error as a quantitative measure. ϵ-error applies for datasets where there is no ground truth age but instead a group of people guessing the ground truth. It takes into account the standard deviation σ of the age voted by the people who labeled the images. Thus if the labeled age for an image varies significantly among the votes, a wrong prediction is penalized less. By assuming that those votes are following a normal distribution with mean age μ and standard deviation σ the error is then measured as

$$\epsilon = 1 - e^{-\frac{(x-\mu)^2}{2\sigma^2}}. \tag{4.1}$$

The final ϵ-error is the average over all images. Its value ranges from 0 (perfect predictions) to 1 (completely wrong).

4.3.4 *Output layer and expected value*

The pre-trained CNN (with VGG-16 architecture) for the ImageNet classification task has an output layer of 1000 softmax-normalized neurons, one for each of the object classes. In contrast, age estimation is a regression and not a classification problem, as age is continuous rather than a set of discrete classes.

For regression we replace the last layer with only 1 output neuron and employ an Euclidean loss function. Unfortunately training a CNN directly for regression is relatively unstable as outliers cause a large error term. This results in very large gradients which makes it difficult for the network to converge and leads to unstable predictions.

Instead, we phrase the prediction problem as a classification problem where the age values are discretized into $|Y|$ ranges of age. Each age range Y_i covers a range of ages from $Y_i^{\min}$ to $Y_i^{\max}$ and votes for the mean of all training samples in this age range, y_i. In our experiments we consider: a) uniform ranges where each age range covers the same number of years and b) balanced ranges such that each age range covers approximately the same number of training samples and, thus, fit the data distribution. The number

of age ranges depends on the training set size, *i.e.* each age range needs sufficiently many training samples and thus finer discretization requires more samples. In this way, we train our CNN for classification and at test time we compute the expected value over the softmax-normalized output probabilities of the $|Y|$ neurons

$$E(O) = \sum_{i=1}^{|Y|} y_i \cdot o_i, \tag{4.2}$$

where $O = \{1, 2, ..., |Y|\}$ is the $|Y|$-dimensional output layer and $o_i \in O$ is the softmax-normalized output probability of neuron i. Experimental results show that this formulation increases robustness during training and accuracy during testing. Additionally it allows some interpretation of the output probability distribution to estimate the confidence of the prediction, which is not possible when training directly for regression.

4.3.5 *Implementation details*

Depending on the experiment, the CNN is trained for regression or classification. In the case of classification we report both the performance when testing for classification, *i.e.* the predicted age is the age of the neuron with the highest probability, and the expected value over the softmax normalized output probabilities.

When training the CNN for classification instead of regression, the age ranges are formed in two different ways: a) uniform ranges such that each age range covers the same number of years and b) balanced ranges where each age range covers approximately the same number of training samples.

For all experiments the CNN is initialized with the weights from training on ImageNet. This model is then further pre-trained on the IMDB-WIKI dataset for classification with 101 output neurons and uniform age ranges. Finally the CNN is trained on the dataset to test on.

We split the training set into 90% for learning the weights and 10% for validation during the training phase. The training is terminated when then network begins to over-fit on the validation set. All experiments start with the pre-trained ImageNet weights

from [134]. For any fine-tuning the learning rate for all layers except the last layer is set to 0.0001. As we change the number of output neurons, the weights of the last layer are initialized randomly. To allow quick adjustment of those new weights, we set the learning rate for the output layer to 0.001. We train with a momentum of 0.9 and a weight decay of 0.0005. The learning rate is reduced every 10 passes through the entire data by a factor of 10.

The models are trained using the Caffe framework [72] on Nvidia Titan X GPUs. Training on the IMDB-WIKI and CACD datasets took several days whereas fine-tuning on the smaller datasets took only a couple of hours.

4.3.6 *Parameters for output layer*

Both Tab. 4.1 and Fig. 4.3 show how varying the number of output neurons and the prediction of ranges of age affects the performance. For all the settings we use LAP train data for training and report on the LAP validation data. Note that for the case where the settings are kept identical with the IMDB-WIKI pre-training which was done with 101 output neurons and uniform balancing, we additionally report performance for the case when the last layer is reinitialized when training on the LAP dataset. There seems to be a sweet spot, *i.e.* too many neurons result into too little training data per neuron and at the same time too few neurons lack a fine-grained ranges of the ages and thus make prediction less precise. Surprisingly, with 10 output neurons the performance is still very good despite the large distance in age between the neurons. Balanced ranges seems to perform slightly better than uniform ranges, especially when combined with few neurons.

For reference in Tab. 4.1 we report the performance when employing standard Support Vector Regression (SVR) with RBF kernel and ϵ-insensitive loss function on deep features extracted from the last pooling layer (conv5_3), last (fc7) and penultimate (fc6) fully connected layer of our deep architecture without and with pre-training on IMDB-WIKI dataset. As expected the specialized layers lead to better performance than the more generic pooling layer when the network is adapted to the age estimation task, otherwise the more generic pooling layer provides better features for

SVR. With IMDB-WIKI pretraining, SVR on fc7 is slightly below the direct application of the network learned for apparent age regression.

4.4 EXPERIMENTS

In this section we present the experimental results. We first introduce the datasets used. In the following we present both quantitative as well as qualitative results. We conclude the section with a discussion about the results.

4.4.1 *Datasets*

In this chapter we use 5 different datasets for real (biological) and apparent age. Fig. 4.4 depicts exemplar images for each dataset. Tab. 4.2 shows the size of each dataset and the corresponding splits for training and testing.

IMDB-WIKI. We introduce a new dataset for age estimation which we name IMDB-WIKI. To the best of our knowledge this is the largest publicly available dataset for age estimation of people in the wild containing more than half a million labeled images. Most face datasets which are currently in use (1) are either small (*i.e.* tens of thousands of images) (2) contain only frontal aligned faces or (3) miss age labels. As the amount of training data strongly affects the accuracy of the trained models, especially those employing deep learning, there is a clear need for large datasets. For our IMDB-WIKI dataset we crawl images of celebrities from IMDb [1] and Wikipedia [2]. For this, we use the list of the 100,000 most popular actors as listed on the IMDb website and automatically crawl from their profiles date of birth, name, gender and all the images related to that person. Additionally, we crawl all profile images from pages of people from Wikipedia with the same meta information.

[1] www.imdb.com
[2] en.wikipedia.org

For both data sources we remove the images that do not list the year when it was taken in the caption. Assuming that the images with single faces are likely to show the celebrity and that the year when it was taken and date of birth are correct, we are able to assign to each such image the biological (real) age. Especially the images from IMDb often contain several people. To ensure that we always use the face of the correct celebrity, we only use the photos where the second strongest face detection is below a threshold. Note that we can not vouch for the accuracy of the assigned age information. Besides incorrect captions, many images are stills from movies - movies that can have extended production times. Nonetheless for the majority of the images the age labels are correct. In total IMDB-WIKI dataset contains 523,051 face images: 460,723 face images from 20,284 celebrities from IMDb and 62,328 from Wikipedia. Only 5% of the celebrities have more than 100 photos, and on average each celebrity has around 23 images.

The dataset is publicly available at
http://data.vision.ee.ethz.ch/cvl/rrothe/imdb-wiki/.
We also released pre-trained models. Note that this dataset can also be used for gender classification. We provide the entire image, the location of the face, its score and the score of the second most confident face detection.

FG-NET. The Face and Gesture Recognition Research Network (FG-NET) [111] aging database consists of 1002 color and gray-scale images which were taken in a totally uncontrolled environment. On average there are 12 images for each of the 82 subjects, whose age ranges from 0 to 69. For evaluation we adopt the setup of [16], [20], [58], [168]. They use leave-one person-out (LOPO) cross validation and report the average performance over the 82 splits.

MORPH. The Craniofacial Longitudinal Morphological Face Database (MORPH) [118] is the largest publicly available longitudinal face database containing more than fifty thousand mug shots. For our experiments we adopt the a setup often used in the literature [16], [20], [58], [128], [159], where a subset of 5,475 photos is used whose age ranges from 16 to 77. For evaluation, the dataset is randomly divided into 80% for training and 20% for testing. Some

works [59], [70] use different splits. We still report them, however they are not directly comparable.

CACD. The Cross-Age Celebrity Dataset (CACD) [17] contains 163,446 images from 2,000 celebrities collected from the Internet. The images are collected from search engines using celebrity name and year (2004-2013) as keywords. The age is estimated using the query year and the known date of birth. The dataset splits into 3 parts, 1800 celebrities are used for training, 80 for validation and 120 for testing. The validation and test sets are cleaned whereas the training set is noisy. In our experiments we report results on the test set.

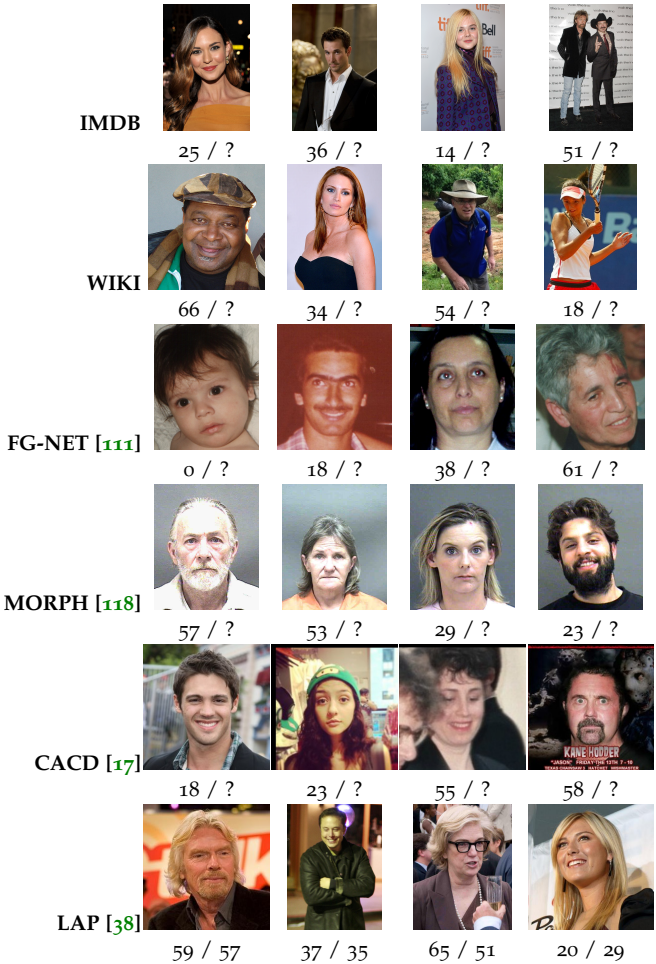
LAP. The ChaLearn LAP dataset [38] contains 4699 images collectively age labeled using two web-based applications. According to the organizers of the LAP challenge this is the largest dataset on apparent age estimation. Each age label is the averaged opinion of at least 10 independent users. Additionally, the standard deviation σ is also provided for each age label. The LAP dataset is split into 2476 images for training, 1136 images for validation and 1087 images for testing. The age distribution is very similar in all the three sets of the LAP dataset. Regarding the distribution of ages, the LAP datasets covers the $20-40$ years interval the best. For the $[0, 15]$ and $[65, 100]$ age intervals it suffers from a small number of images per year.

As Fig. 4.5 depicts, the distribution of age between the datasets differs greatly. FG-NET contains images with by far the youngest people. MORPH has 2 peeks, one around early 20s and one at 40, suggesting that the images come from two data sources. CACD has few images from people below 20 or above 60 but is very balanced between those ages. The majority of face images on Wikipedia seem to show people slightly younger than on IMDb. In contrast Wikipedia has a long tail for old ages. The combined IMDB-WIKI dataset then follows a very similar distribution to the IMDb dataset as the ratio between IMDB and WIKI images is about 8 to 1. LAP and WIKI datasets have similar distributions.

Table 4.2: The proposed method is evaluated on 5 datasets. This table shows the number of images per dataset and the corresponding training and testing split.

Dataset	Number of images
IMDB-WIKI	**523,051**
IMDB	460,723
Wikipedia	62,328
IMDB-WIKI Train	260,282
FG-NET [111]	**1,002**
Train	990 (average)
Test	12 (average)
MORPH [118]	**55,134**
Train	4,380
Test	1,095
CACD [17]	**163,446**
Train	145,275 (1,800 celebs)
Val	7,600 (80 celebs)
Test	10,571 (120 celebs)
LAP [38]	**4,691**
Train	2,476
Val	1,136
Test	1,079

Figure 4.4: Real / Apparent age of exemplar images for each dataset



4.4.2 *Quantitative results*

In this section we report quantitative results of our proposed DEX method for biological and apparent age estimation. Additionally

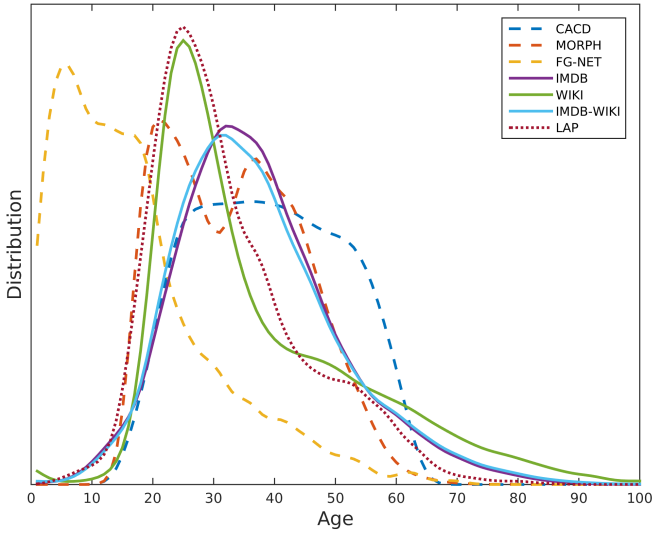


Figure 4.5: Age distribution of people for all 5 datasets.

the results from the ChaLearn Looking at People (LAP) 2015 challenge [38] on apparent age estimation are presented.

Apparent age estimation

We report performance of our DEX method for apparent age estimation. Tab. 4.1 summarizes the results when testing on the validation set of the LAP dataset.

The best performance for pre-training on the IMDB-WIKI dataset and taking the expected value reaches 0.282 ϵ-error (MAE 3.252) compared to 0.456 ϵ-error (MAE 5.369) when training directly on the LAP dataset. Training for regression instead performs worse at 0.475 ϵ-error (MAE 5.586) and 0.310 ϵ-error (MAE 3.650), respectively.

Looking at people (LAP) challenge

Our DEX method is the winner of the ChaLearn Looking at People (LAP) 2015 challenge [38] on apparent age estimation with 115 registered teams, significantly outperforming the human reference. The challenge had two phases: development and test.

Development phase. In this phase the training and validation images of the LAP dataset are accessible. For the training set the apparent age labels are known, whereas for the validation set they are not released. The teams are able to submit their predictions for the validation images to a server to get the overall performance on those images. A public score board shows the latest performance of each team. As the previous score of each team is overwritten we build a crawler to check the score board every couple of seconds. Fig. 4.6 shows the scores over the last month of the development phase. It can be clearly seen that as the end of the phase approaches the teams steadily improve their performance.

Test phase. This is the final phase of the competition. The organizers of the challenge release the apparent age labels for the validation set and the images for the final test set. Now the algorithm is re-trained on both training and validation images to then predict the apparent age on the final test set. Our final results are obtained by training a full ensemble of 20 CNNs with 101 age bins on the training and validation images and then averaging the 20 predictions for each of the test images. Note that for all other results in this chapter we report the performance of a single CNN.

Final results. Fig. 4.3 shows the final ranking of the competition. The best 4 methods achieve performance above the human reference of an ϵ-error of 0.34, as reported by the organizers. Our method is the only method within the top 6 methods which does not employ facial landmarks.

Real age estimation

In this section we present the performance of our proposed method for estimating the real (biological) age. In recent years, both the FG-

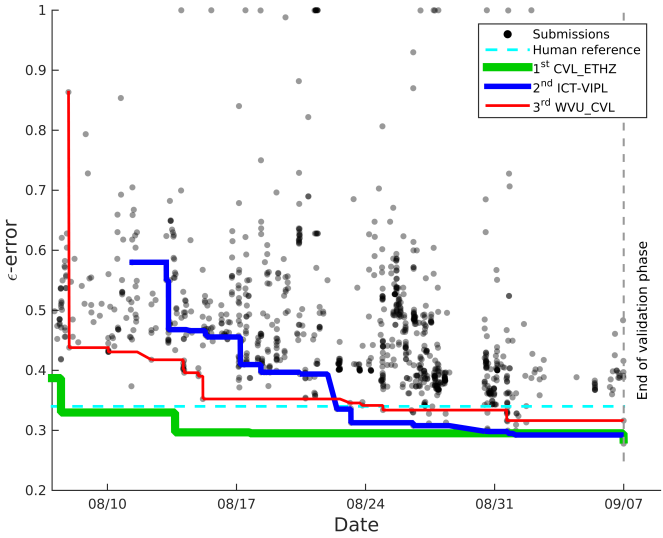


Figure 4.6: One month validation entries for LAP challenge. For the top 3 teams we plot the best scores curves. CVL_ETHZ is ours.

Table 4.3: ChaLearn LAP 2015 [38] final ranking on the test set. 115 registered participants. AgeSeer does not provide codes. The human reference is the one reported by the organizers.

Rank	Team	ϵ error
1	CVL_ETHZ (ours) [125]	0.264975
2	ICT-VIPL [96]	0.270685
~~3~~	~~AgeSeer~~	~~0.287266~~
3	WVU_CVL [184]	0.294835
4	SEU-NJU [171]	0.305763
	human reference	*0.34*
5	UMD	0.373352
6	Enjuto	0.374390
7	Sungbin Choi	0.420554
8	Lab219A	0.499181
9	Bogazici	0.524055
10	Notts CVLab	0.594248

NET and MORPH dataset have become the standard benchmark for the existing methods.

On the MORPH dataset, our DEX method achieves a mean average error (MAE) of 3.25 when just fine-tuning the CNN on the training MORPH data. This improves over previous state-of-the-art reported in [128] by 0.2 years (see Tab. 4.4). Additional fine-tuning on our novel IMDB-WIKI dataset before fine-tuning on the MORPH dataset leads to a MAE of 2.68 years. To the best of our knowledge this is the first work reporting an error below 3 years on this common evaluation setup for MORPH, improving over the state-of-the-art by nearly 0.8 years.

On the FG-NET dataset, without fine-tuning on IMDB-WIKI we achieve 4.63 years. Note that the larger error is due to the fact that FG-NET is a very small dataset (1000 images) and thus training a CNN on it is difficult. However, training on the IMDB-WIKI dataset before fine-tuning on FG-NET leads to a MAE of 3.09 years. This improves over DLA [159] by more than 1 year in average error. The results are summarized in Tab. 4.4.

Table 4.4: Comparison results (MAE) for real (biological) age estimation. Our DEX method achieves the state-of-the-art performance on the MORPH and FG-NET standard datasets (*different split, **landmark pre-training).

Method	MORPH 2 [118]	FG-NET [111]
Human workers [63]	6.30	4.70
DIF [63]	3.80*	4.80
AGES [51]	8.83	6.77
MTWGP [182]	6.28	4.83
CA-SVR [20]	5.88	4.67
SVR [58]	5.77	5.66
OHRank [16]	5.69	4.85
DLA [159]	4.77	4.26
[70]	4.25*	N/A
[61]	4.18*	N/A
[59]	3.92*	N/A
[97]	N/A	4.37*
[98]	N/A	4.12**
[175]	3.63*	N/A
[128]	3.45	5.01
DEX	3.25	4.63
DEX (IMDB-WIKI)	**2.68**	**3.09**

On the CACD dataset [17] we run additional experiments. The results are shown in Tab. 4.5. In comparison to MORPH and FG-NET the CACD dataset is much larger but not manually annotated. When training on the 145,275 training images we achieve a MAE of 4.785 years. When only training on the manually cleaned validation set with 7600 images the performance drops to a MAE of 6.521. This suggests that having a large training set with slightly imprecise labels results in better performance than a carefully annotated dataset of much smaller size.

Age group estimation

Besides real age estimation, we also evaluate our approach for predicting age groups. This is a somewhat simpler task as the goal is

Table 4.5: DEX results (MAE) on CACD dataset.

Training on	CACD [17]
Train	4.785
Val	6.521

to predict whether a person's age falls within some range instead of predicting the precise biological age. We evaluate the performance on the Adience dataset [36] which consists of 26,580 images from 2,284 subjects from Flickr. The dataset has 8 age groups (0-2, 4-6, 8-13, 15-20, 25-32, 38-43, 48-53, 60- years) and we report the results on the 5-fold cross validation proposed by the authors of the dataset. For this task we train our network for classification with 8 classes and report the exact accuracy (correct age group predicted) and 1-off accuracy (correct or adjacent age group predicted). We report results with and without pre-training on IMDB-WIKI. As it can be seen in Tab. 4.6 we achieve an accuracy of 64.0% compared to the previous state-of-the-art of 50.7%. When predicting the 1-off accuracy we achieve 96.6%, *i.e.* our model is nearly always able to predict at least the adjacent age group.

Table 4.6: Age group estimation results (mean accuracy [%] $\pm$ standard deviation) on Adience benchmark [36].

Method	Exact	1-off
DEX w/ IMDB-WIKI pretrain	**64.0 $\pm$ 4.2**	**96.6 $\pm$ 0.9**
DEX w/o IMDB-WIKI pretrain	55.6 ± 6.1	89.7 ± 1.8
Best from [92]	50.7 ± 5.1	84.7 ± 2.2
Best from [36]	45.1 ± 2.6	79.5 ± 1.4

4.4.3 *Insight experiments*

In the following we present various insight experiments. These experiments are both quantitative and qualitative and give a deeper understanding of the method.

Visual assessment. Fig. 4.7 shows examples of face images in the wild (from LAP dataset) with good age estimation by our DEX

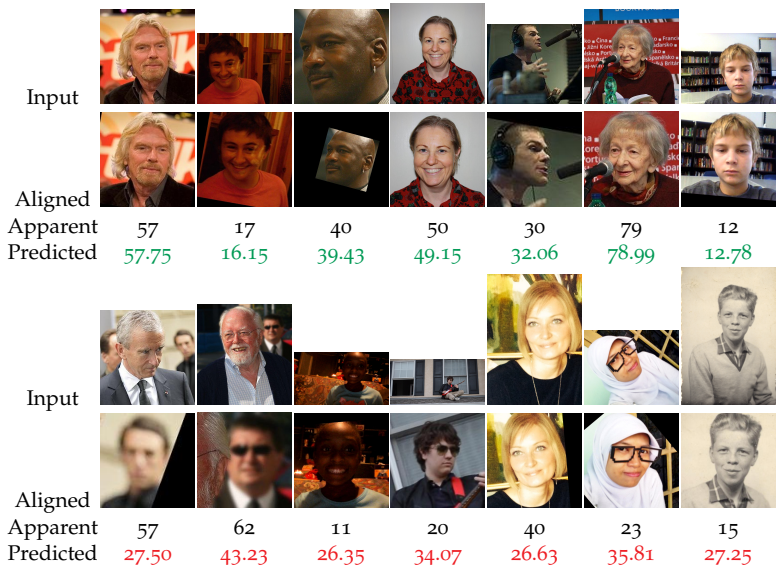
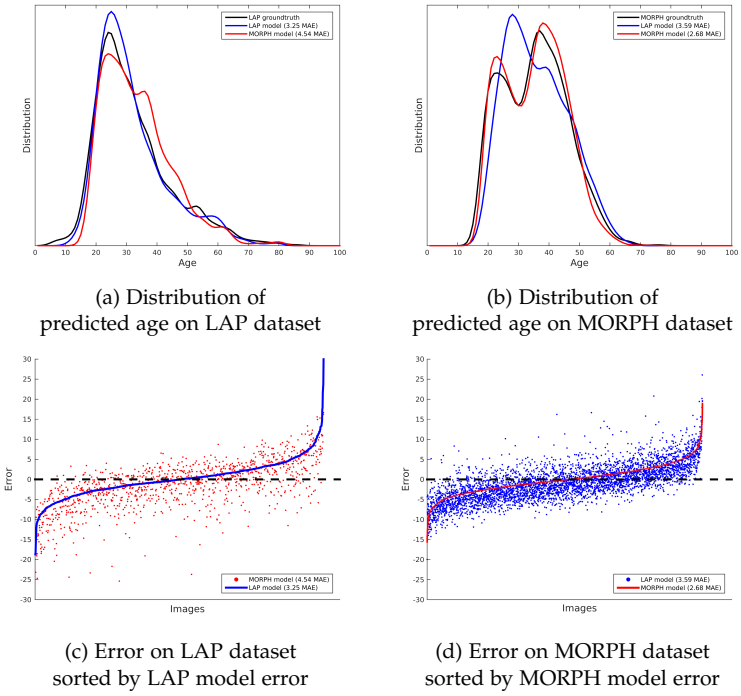


Figure 4.7: Examples of face images with good and bad age estimation by DEX.

with a single CNN. We observe that in these cases also the faces are aligned very well. Failure cases are also shown in Fig. 4.7. The failures are mostly caused by a failure in the detection stage (*i.e.* wrong or no face detected) or difficult conditions due to glasses, other forms of occlusions, or bad lightning.

Dataset bias. In Fig. 4.8 we reveal the existence of a dataset bias. By testing the trained models on a dataset other than it was trained for (trained on LAP and tested on MORPH, and vice versa) we show the biases which come with each dataset. In Fig. 4.8 (a) we show the distribution of predicted ages on LAP dataset for two models trained on MORPH dataset and LAP dataset, resp., and of the LAP dataset. The LAP model follows the distribution of the dataset and has the better MAE. In contrast the MORPH model exhibits a bi-modal distribution which is more similar to the MORPH dataset (cf. Fig. 4.8 (b)). A similar behavior is observed when testing both models on the MORPH dataset (see Fig. 4.8 (b)). In Fig. 4.8 (c,d) the individual errors for each test image are plotted. The im-

ages are sorted according to the original dataset, *i.e.* in Fig. 4.8(c) when testing on LAP they are sorted according to the error of the model trained on LAP. On LAP dataset, in Fig. 4.8(c), it can be seen that even though the error of the MORPH model is bigger overall, its predictions follow the curve of the LAP trained model and thereby both models similarly over- or underestimate the age of a person. A similar reasoning applies to the plots in Fig. 4.8(d).



(a) Distribution of predicted age on LAP dataset

(b) Distribution of predicted age on MORPH dataset

(c) Error on LAP dataset sorted by LAP model error

(d) Error on MORPH dataset sorted by MORPH model error

Figure 4.8: Dataset bias of LAP and MORPH.

Important face regions. In order to determine which parts of a face image correlate and contribute the most to the overall age estimation accuracy we devise the following experiment. We systematically occlude a vertical or horizontal strip of the image by setting it to the mean image, as in [179]. Each of the 20 strips has a width of 10% of the input face image. In Fig. 4.10 we report

the MAE on the LAP dataset (validation images) for each of the vertical or horizontal strip occlusions. The results are intuitive, occlusions in the face area from the eyes to the chin and between ears affect the most the estimation accuracy. The results show that occluding the eyes with a horizontal strip increases the MAE the most, suggesting that the eyes are the most important indicator for age in the human face. The eyes are seconded by the horizontal strip region passing the upper lip and bottom of the nose. At the same time the horizontal strip occlusions lead to larger MAE than the vertical ones. A reason for this is that the face has horizontal symmetry and therefore for vertical occlusions except the strip that passes through the center of the face, there is always a corresponding symmetrical strip that is not occluded providing important information to the CNN model.

Robustness to block occlusions. To determine the robustness of our solution to occlusion we apply a block occlusion mask at random locations in the input face image. We report the MAE over the LAP dataset as the size of the occluded area is increased in Fig. 4.11. When less than 20% of the image is occluded the MAE is still low, *i.e.* the trained CNN is robust to those fairly small occlusions. Above 40% occlusion the MAE performance rapidly deteriorates.

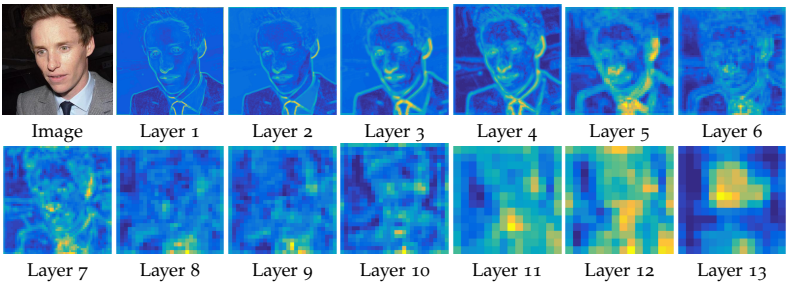


Figure 4.9: Activation across CNN for a test image. The color indicates the maximum activation for any feature map for a particular layer.

CNN model visualization. Fig. 4.12 shows a t-Distributed Stochastic Neighbor Embedding (t-SNE) [99] of the last fully connected

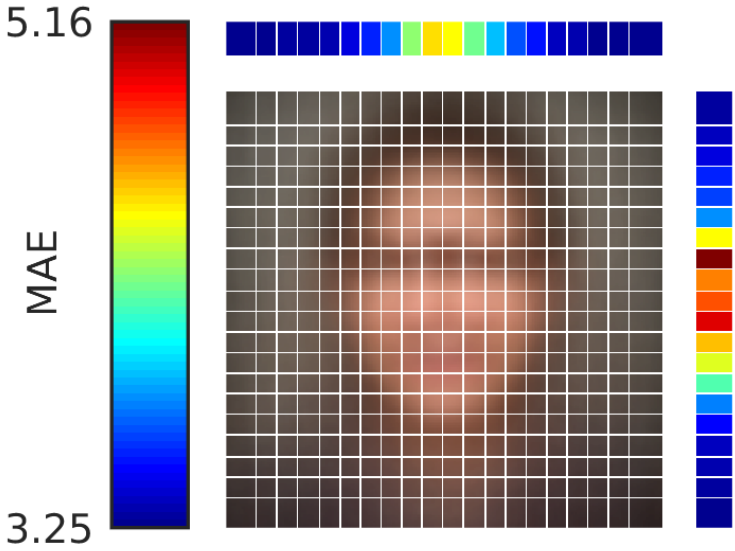


Figure 4.10: Systematic occlusion of horizontal and vertical strips on test images and its impact on the MAE (inspired by [179]).

layer of the model trained on the LAP dataset for the validation images. The feature vector of dimensionality 4096 is preprocessed using PCA to a dimensionality of 50. The visualization shows the test images for a perplexity of 10. We further cluster the embedded data into 20 clusters and report the average age of each cluster. The separation of images by age suggests that the features learned are discriminative for age prediction.

CNN activations. Fig. 4.9 shows the activation across our CNN trained on LAP for a test image using a color heatmap. The color indicates the maximum activation energy for any feature map for a particular layer. In the first couple of layers the face of the person can still be recognized and we can generally have the intuition that the neurons corresponding to the face region and the face edges activate the most. However, as we go deeper into the CNN the representation becomes more abstract and difficult to interpret.

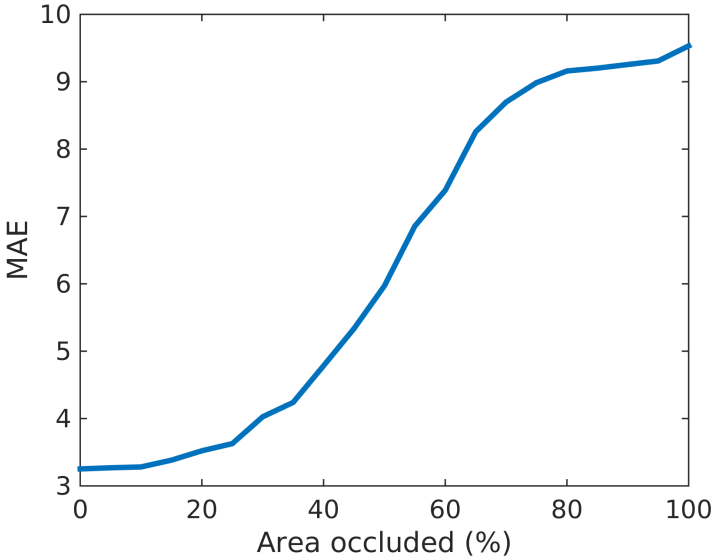


Figure 4.11: Impact of random occlusion of test image on the performance (MAE).

4.4.4 *Discussion*

The proposed DEX method shows state-of-the-art results on MORPH and FG-NET for biological age and LAP for apparent age. Training the CNN for classification instead of regression not only improves performance but also stabilizes the training process. Without relying on landmarks and by robustly handling small occlusions the proposed method confirms its applicability for age estimation in the wild. Pre-training on the IMDB-WIKI dataset results in a large boost in performance suggesting that the lack of a larger dataset for age estimation was overdue for a long time.

In future work the training dataset could be further enlarged. Fine-tuning the face detector on the target dataset can reduce the failure rate of the face detection step. Using a very robust landmark detector can lead to better alignment. The recently introduced Residual Nets by [65] with more than 150 layers show that

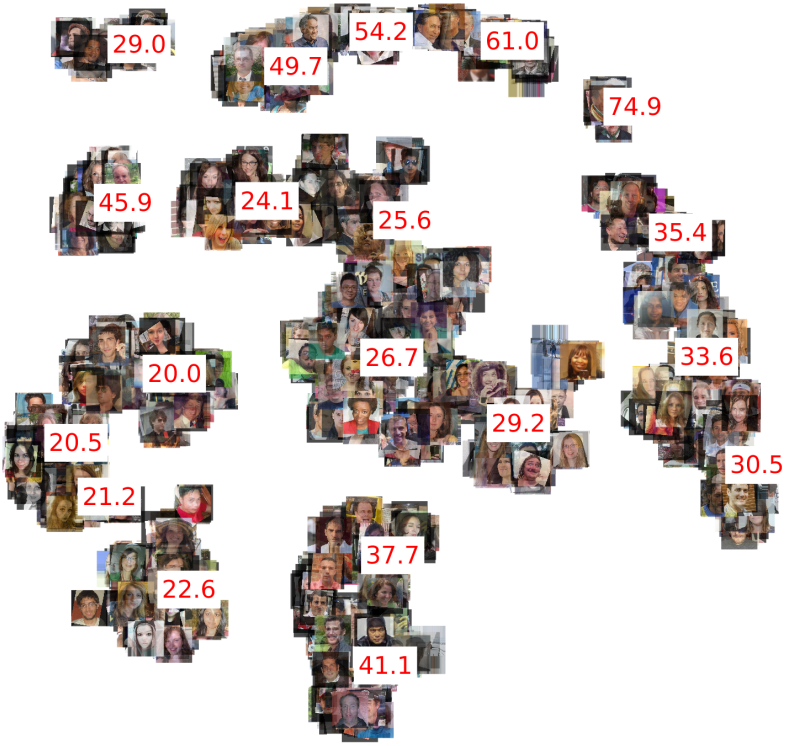


Figure 4.12: t-SNE embedding and average age per cluster.

an even deeper architecture than VGG-16 might help to improve performance if sufficient training data is available. Though at the same time the work suggests that there is an optimal depth, as the network with 1000 layers performs worse.

Ultimately the proposed DEX pipeline can be used for other prediction tasks of facial features including gender, ethnicity, attractiveness or attributes (*i.e.* does the person have glasses, a beard, blond hair).

4.5 CONCLUSIONS

In this chapter we proposed a solution for real and apparent age estimation. Our Deep EXpectation (DEX) formulation builds upon a robust face alignment, the VGG-16 deep architecture and a classification followed by a expected value formulation of the age estimation problem. Another contribution is IMDB-WIKI, the largest public face images dataset to date with age and gender annotations. We validate our solution on standard benchmarks and achieve state-of-the-art results.

In the next chapter we go beyond predicting objective facial attributes such as age and present a framework to predict facial attractiveness, which is highly subjective.

5 VISUAL GUIDANCE FOR PREFERENCE PREDICTION

5.1 INTRODUCTION

'First impressions count' the saying goes. Indeed, psychology confirms that it only takes 0.1 seconds for us to get a first impression of someone [161], with the face being the dominant cue. Factors that are relevant for survival seem the ones evolution makes us pick up the fastest. These include age, gender, and attractiveness. We will call those quantifiable properties, such as age and gender, 'demographics'.

An everyday demonstration is that people on dating sites often base their decisions mainly on profile images rather than textual descriptions of interests or occupation. Our goal of this chapter is to let a computer predict someone's preferences from single facial photos (in the wild). In particular, we try to predict how attractive a previously unseen face would be for a particular person who has already indicated preferences for people in the system. This goes beyond just predicting objective facial attributes as in the previous chapter, since attractiveness is highly subjective.

Our main benchmark is a large dataset of more than 13,000 user profiles from a dating site. We have access to their age and gender, as well as more than 17 million 'hot or not' ratings by some users about some other users (their profile photo). The ratings are very sparse when compared to their potential number. For people who have given ratings, we want to predict new ratings for other people in and outside the dataset.

The visual information, here the profile image, presumably containing a face, is the main basis for any user-to-user rating. Therefore, we employ a face detector and crop the best detected face and its surrounding context (corresponding to body and posture) from which we extract deep features by means of a (fine-tuned) convolutional neural network. In order to make sure that these features

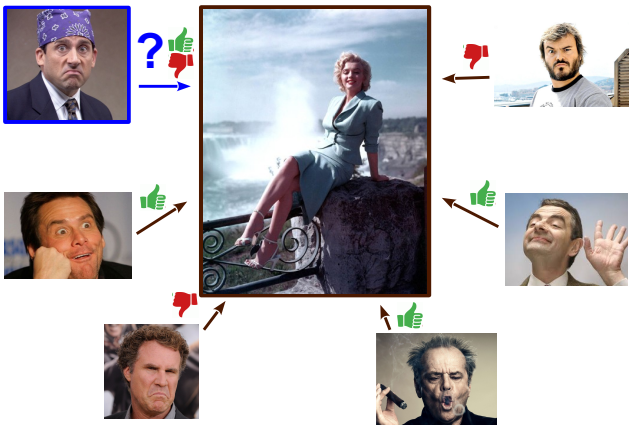


Figure 5.1: Can we infer preferences from a single image?

are appropriate for the main task – automated attractiveness rating - we first test our features on age, gender, and facial beauty estimation for which previous methods and standard datasets exist. We obtain state-of-the-art results.

For predicting preferences for users with known ratings for a subset of others in the dataset, collaborative filtering is known to provide top results, *i.e.* for movie [76] or advertisement suggestions [120]. We adapt this framework to take account of visual information, however. As our experiments will show, adding visual information improves the prediction, especially in cases with few ratings per user. In case of a new face, not part of the dataset and thus without a history of preferences, we propose to regress the input image to the latent space of the known users. By doing so, we alleviate the need for past ratings for the query and solely rely on the query image.

The same technique can be applied to different visuals-enhanced tasks, such as rating prediction of movies, songs, shopping items, in combination with a relevant image (*e.g.* movie poster, CD cover, image of the item). We test on the MovieLens dataset augmented with poster images for each movie, a rather weak information, to demonstrate the wider applicability of our approach.

We demonstrate our algorithms on howhot.io, a website where people can upload a photo of their face and an algorithm will then estimate the age, gender and facial attractiveness of the person.

The main contributions of this chapter are:

- an extensive study on the inference of information from profile images using the largest dating dataset thus far
- a novel collaborative filtering approach that includes visual information for rating/preference prediction
- a novel regression technique for handling visual queries without rating history which prior work cannot cope with

5.2 RELATED WORK

The focus of this chapter is to infer as much information as possible from a single image and to predict subjective preferences based on an image query with possibly a prior rating history. Next we review related works.

Image features. Instead of handcrafted features like SIFT, HoG, or Gabor filters, we use learned features obtained using neural networks [52], [77], [134]. The latter have shown impressive performance in recent years. Such features have already been used for age and gender estimation in the previous chapter or in [125], [159].

Demographics estimation. Multiple demographic properties such as age, gender, and ethnicity have been extracted from faces. A survey on age prediction is provided by Fu *et al*. [46] and on gender classification by Ng *et al*. [108]. Kumar *et al*. [79] investigate image 'attribute' classifiers in the context of face verification. Some approaches need face shape models or facial landmarks [62], [71], others are meant to work in the wild [16], [20], [125], [159] but still assume face localization. Generally, the former approaches reach better performance as they use additional information. The errors in model fitting or landmark localization are critical. Moreover, they require supervised training, detailed annotated datasets, and higher computation times. On top of the extracted image features a machine learning approach such as SVM [152] is employed to learn a demographics prediction model which is then applied to

new queries. For more related work on age estimation, see Chapter 4.

Subjective property estimation. While age and gender correspond to objective criteria, predicting the attractiveness of a face is more subjective. Nonetheless, facial beauty [4], [37], [56], [106], [167] can still be quantified by averaging the ratings by a large group. Benchmarks and corresponding estimation methods have been proposed. In the subjective direction, Dhar *et al*. [30] demonstrate the aesthetic quality estimation and predict what they call 'interestingness' of an image, while Marchesotti *et al*. [102] discover visual attributes (including subjective ones) to then to use them for prediction. Also, recently Kiapour *et al*. [75] inferred complex fashion styles from images. Generally, the features and methods used for age and gender can be adapted to subjective property estimation, and we do the same in this chapter. From the literature we can observe three trends: (i) besides Whitehill and Movellan [160], most papers focus on predicting facial beauty averaged across all ratings, whereas we aim at predicting the rating by a specific person; (ii) as pointed out in the study by Laurentini and Bottino [86] usually small datasets are used, sometimes with less than 100 images and with only very few ratings per image – our dataset contains more than 13,000 images with more than 17 million ratings; (iii) most datasets are taken in a constrained environment showing aligned faces. In contrast, our photos contain in many cases also parts of the body and some context in the background. Thus, we focus not just on facial beauty but on general attractiveness of the person – referred to as hotness in the following.

Preferences/ratings prediction. The Internet brought an explosion of choices. Often, it is difficult to pick suitable songs to hear, books to read, movies to watch, or - in the context of dating sites - persons to contact. Among the best predictors of interest are collaborative filtering approaches that use the knowledge of the crowd, *i.e*. the known preferences/ratings of other subjects [11], [131]. The more prior ratings there are, the more accurate the predictions become. Shi *et al*. [133] survey the collaborative filtering literature. Matrix factorization lies at the basis of many top collaborative filtering methods [76], [91]. Given the importance of the visual information in many applications, we derive a matrix factorization formulation

regularized by the image information. While others [133] proposed various regularizations, we are the first to prove that visual guidance helps preference prediction. Moreover, we propose to regress queries without rating history to a latent space derived through matrix factorization for the known subjects and ratings.

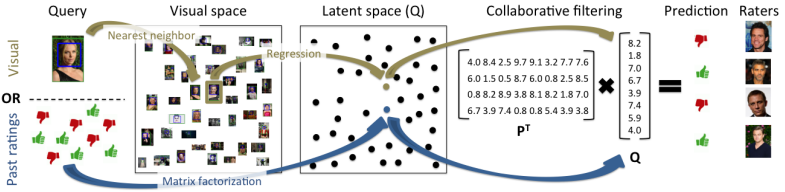


Figure 5.2: Preferences prediction scheme. For a visual query without past ratings we first regress to the latent Q space (obtained through matrix factorization) to then obtain the collaborative filtering prediction as in the case for the queries with known past ratings and Q factor.

Social networks. The expansion of the Internet and the advance of smartphones boosted the (online) social networks worldwide. Networks such as Facebook facilitate interaction, sharing, and display of information and preferences among individuals. Yet, time is precious and hence efforts are made to develop filters and ranking tools to assist users. A recent study by Youyou *et al.* [176] shows that accurate predictions about the personality of a user can be made using her/his 'likes'. Contents and ads can then be personalized and this is extremely important for social networks and search engines such as Google [135]. This chapter focuses on dating sites and the prediction of attractiveness ratings. Most such works [12], [78] rely on past ratings and cannot cope when there are none or few.

5.3 VISUAL FEATURES

Razavian *et al.* [116] showed that features extracted from convolutional neural networks (CNN) are very powerful generic descriptors. Inspired by that, for all our experiments we use the VGG-

16 [134] features which are pre-trained on a large ImageNet object dataset and result in a descriptor of length 4,096. We use the implementation by Vedaldi and Lenc [154]. We reduce the dimensionality using PCA to keep $\sim$ 99% of the energy. Before we use these feature to predict attractiveness to a particular user, we first confirm that the extracted visual features are powerful enough to capture minor facial differences by predicting age and gender.

We perform reference experiments on a widely used dataset for age prediction, the MORPH 2 database [118]. We also test gender estimation on the same MORPH 2 dataset. Unfortunately, besides the dataset provided by Gray *et al*. [56] – to the best of our knowledge – there are no other publicly available large datasets on averaged facial beauty. As shown next our features achieve state-of-the-art performance for age, gender, and facial beauty prediction. We believe that this good performance is mostly due to the depth of the model with 16 layers, compared with previous state-of-the-art using only 6 layers [159].

5.3.1 *Predicting age and gender*

Our experiments are conducted on a publicly available dataset, the MORPH 2 database [118]. We adopt the experimental setup of [16], [20], [58], [159], where a set of 5,475 individuals is used whose age ranges from 16 to 77. The dataset is randomly divided into 80% for training and 20% for testing. Following the procedure described in [52], our CNN features are fine-tuned on the training set.

The age is regressed using Support Vector Regression (SVR) [15] with an RBF kernel and its parameters are set by cross-validation on a subset of the training data. We report the performance in terms of mean absolute error (MAE) between the estimated and the ground truth age.

As shown in Table 5.1, we achieve state-of-the-art performance on the MORPH 2 dataset (3.45 years MAE) by reducing the error of the currently best result (4.77 MAE reported by [159]) with more than a full year. For gender prediction on the MORPH 2 dataset we keep the same partition as for age and achieve 96.26% accuracy, which, despite the small training set, is on par with other results in the literature [60], [61]. Fig. 5.4 shows several good and erroneous

Table 5.1: Age estimation performance in terms of mean absolute error (MAE) on the MORPH 2 dataset. We improve the state-of-the-art results by more than 1 year.

Method	MORPH 2 [118]
AGES [51]	8.83
MTWGP [182]	6.28
CA-SVR [20]	5.88
SVR [58]	5.77
OHRank [16]	5.69
DLA [159]	4.77
Proposed Method	**3.45**

Figure 5.3: Average faces for 5 clusters based on age or beauty, resp. Average beauty is less meaningful, suggesting personalized prediction.

predictions of our method on the MORPH 2 dataset. Fig. 5.3 shows averages of faces ranked according to age on MORPH 2 and beauty on Gray, resp. On our more challenging Hot-or-Not dataset (Section 5.5.1) we achieve 3.61 MAE for age and 88.96% accuracy for gender prediction. Note that Chapter 4 provides more in-depth experiments on age estimation. The experiments in this chapter are not more than a proxy to show that the deep features extracted from the face are good at capturing subtle facial details.

5.3.2 *Predicting facial beauty*

Following a similar procedure as for age prediction, we test our features on the dataset introduced by Gray *et al.* [56]. The Gray dataset contains 2056 images with female faces collected from a popular social/dating website[1]. The facial beauty was rated by 30

[1] http://hotornot.com

subjects and the ratings were then normalized as described in [56]. The dataset is split into 1028 images for training and 1028 for testing. We report the average performance across exactly the same 5 splits from the reference paper in terms of Pearson's correlation coefficient, the metric from the original paper. Also, we report performance with and without face alignment using the same alignment algorithm of Huang *et al*. [69].

Table 5.2: Facial beauty estimation performance on Gray dataset with and without face alignment in terms of correlation.

Method	Correlation w/o alignment	Correlation w/ alignment
Eigenface	0.134	0.180
Single Layer Model [56]	0.403	0.417
Two Layer Model [56]	0.405	0.438
Multiscale Model [56]	0.425	0.458
Proposed Method	**0.470**	**0.478**

As shown in Table 5.2 our proposed features achieve state-of-the-art performance on predicting facial beauty as averaged over multiple raters. We improve by more than 10% over the best score reported by [56] for the raw images. A couple of per image results are depicted in Fig. 5.4.

5.4 PREDICTING PREFERENCES

Our goal is to make personalized predictions, such as how a specific male subject $m \in M$ rates a female subject $f \in F$. The rating R_{mf} is 1 if 'm likes f', -1 if 'm dislikes f', and 0 if unknown. f is also called the query user, as at test time we want to predict the individual ratings of all men for that woman. Due to space limitations, we derive the formulation for this case. Yet it is also valid when swapping sexes, *i.e*. when women are rating men.

In the following section we phrase the problem as a collaborative filtering problem, assuming that we know past ratings for both men and women. In Section 5.4.2 we extend the formulation to also consider the visuals of the subjects being rated. In Section 5.4.3 we present a solution to predict the ratings solely based on the visual

information of the subjects, without knowing how they were rated in the past.

5.4.1 *Model-based collaborative filtering (MF)*

We phrase the problem of a user m rating the image of user f as a model-based collaborative filtering problem. The model learned from known ratings is then used to predict unknown ratings. In its most general form, we have

$$g(P_m, Q_f) \Rightarrow R_{mf},\ m = 1, 2, ..., M,\ f = 1, 2, ..., F, \tag{5.1}$$

where the function g maps the model parameters to the known ratings. P_m denotes a set of model parameters describing the preferences of user m. Similarly, Q_f describes the appearance of user f, *i.e.* a low-dimensional representation of how the appearance of a user is perceived. We now estimate the model parameters given the ratings we know.

In recent years, Matrix Factorization (MF) techniques have gained popularity, especially through the Netflix challenge, where it achieved state-of-the-art performance [76]. The basic assumption underlying MF models is that we can learn low-rank representations, so-called latent factors, to predict missing ratings between user m and image f. One can approximate the ratings as

$$R \approx P^T Q = \hat{R}. \tag{5.2}$$

In the most common formulation of MF [133] we can then frame the minimization as

$$\begin{aligned} P^*, Q^* = \underset{P,Q}{\operatorname{argmin}} & \tfrac{1}{2} \textstyle\sum_{m=1}^{M} \sum_{f=1}^{F} I_{mf} (R_{mf} - P_m^T Q_f)^2 \\ & + \tfrac{\alpha}{2} (\|P\|^2 + \|Q\|^2) \end{aligned} \tag{5.3}$$

where P and Q are the latent factors and P^* and Q^* their optimal values. I_{mf} is an indicator function that equals 1 if there exists a rating R_{mf}. The last term regularizes the problem to avoid overfitting.

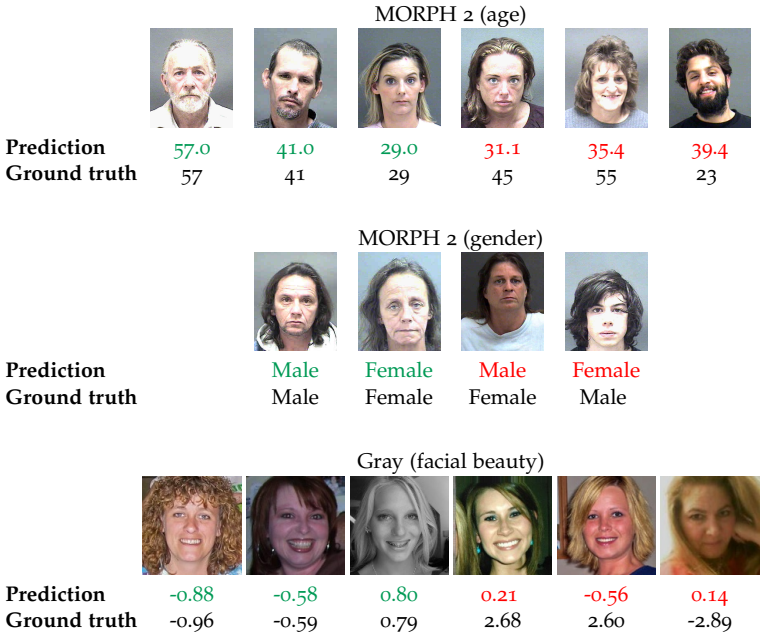


Figure 5.4: Examples of accurately and wrongly predicted age, gender, and facial beauty for the MORPH 2 and Gray datasets.

5.4.2 *Visual regularization (MF+VisReg)*

Knowing that the users in the app rate the subjects of the opposite sex solely based on the image, we make the assumption that people with similar visual features have similar latent appearance factors Q. Thus we can extend the formulation by adding the visual features V of the query images to further regularize the optimization

$$\begin{aligned} L(P,Q) = \; & \tfrac{1}{2}\sum_{m=1}^{M}\sum_{f=1}^{F} I_{mf}(R_{mf} - P_m^T Q_f)^2 \\ & + \tfrac{\alpha_1}{2}(\|P\|^2 + \|Q\|^2) \\ & + \tfrac{\alpha_2}{2}\sum_{f=1}^{F}\sum_{g=1}^{F}(S_{fg} - Q_f^T Q_g)^2. \end{aligned} \tag{5.4}$$

The visual similarity is defined as

$$S_{fg} = \frac{V_f^T V_g}{\left\|V_f\right\|\left\|V_g\right\|}. \tag{5.5}$$

Visually this proves to be a good metric for visual similarity. The optimal latent factors are calculated by gradient descent, where the derivatives are

$$\begin{aligned} \frac{\partial L}{\partial P_m} &= \sum_{f=1}^{F} I_{mf}(P_m^T Q_f - R_{mf})Q_f + \lambda P_m \\ \frac{\partial L}{\partial Q_f} &= \sum_{m=1}^{M} I_{mf}(P_m^T Q_f - R_{mf})P_m \\ &\quad +2\alpha_2 \sum_{g=1}^{F}(Q_f^T Q_g - S_{fg})Q_g + \lambda Q_f. \end{aligned} \tag{5.6}$$

5.4.3 *Visual query*

We now want to predict how user m rates user f without knowing any past ratings of f but knowing her visual feature V_f (see Fig. 5.2). This implies that we do not know the latent factor Q_f for f. The goal is to get an estimate $\hat{Q}_f$ of Q_f based solely on the visual feature V_f. Then we would be able to regress the rating as

$$\hat{R}_{mf} = P_m^T \hat{Q}_f. \tag{5.7}$$

Learning a global regression led to poor results as attractiveness is highly subjective. Instead our approach is inspired by the recently introduced anchored neighborhood regression (ANR) method for image super-resolution [141], where the problem is formulated as a piece-wise local linear regression of low to high resolution image patches and with offline trained regressors. In contrast to ANR, each sample is an anchor and the neighborhood is spanned over all other training samples and weighted according to its similarity to the anchor. This way we are obtaining more robust local regressors that can cope with the scarcity of the data.

As for regularizing MF, we assume that the visual features V and the latent factor Q locally have a similar geometry. Further, we assume that we can locally linearly reconstruct each visual feature or latent factor by its neighbors. Under these assumptions we can reconstruct features and latent factors using the same weights

for the neighbors. In the visual space we now aim to find these weights β by phrasing the problem as a ridge regression

$$\min_{\beta_g} \left\| V_g - N_{V_g}\beta_g \right\|^2 + \lambda \left(\kappa \left\| \Gamma_g \beta_g \right\|^2 + (1-\kappa) \left\| \beta_g \right\|^2 \right), \tag{5.8}$$

where N_{V_g} is a matrix of the neighboring visual features of V_g stacked column-wise and κ is a scalar parameter. The optimization is regularized by the similarity to its neighbors according to eq. 5.5, in the sense that greater similarity yields greater influence on β:

$$\Gamma_g = \text{diag}(1 - S_{g1}, 1 - S_{g2}, ..., 1 - S_{gF}). \tag{5.9}$$

The closed-form solution of the problem can be written as

$$\beta_k = \left[N_{V_g}^T N_{V_g} + \lambda \left[\kappa \Gamma_g^T \Gamma_g + (1-\kappa) I \right] \right]^{-1} N_{V_g}^T V_g. \tag{5.10}$$

As we assume that the latent space behaves similarly locally, we can regress the latent factor Q_g as a linear combination of its neighbors using the same β_g. Note that N_{Q_g} corresponds to the latent factors of N_{V_g}, *i.e.* the neighbors in the visual space. Plugging in our solution for β_g we get

$$\begin{aligned} Q_g &= N_{Q_g}\beta_g \\ &= N_{Q_g}\left[N_{V_g}^T N_{V_g} + \lambda \left[\kappa \Gamma_g^T \Gamma_g + (1-\kappa) I \right] \right]^{-1} N_{V_g}^T V_g \\ &= M_g V_g. \end{aligned} \tag{5.11}$$

Thus we have found a projection M_g from a visual feature V_g to its latent factor Q_g. At test time for a given visual feature V_f, we now aim to find the most similar visual feature in the training space, $\hat{g} = \arg\max_g \ S_{fg}$. Then we use the projection matrix of $\hat{g}$ to obtain $\hat{Q}_f$ to finally estimate the rating of user m for the image of user f as

$$\hat{R}_{mf} = P_m^T \hat{Q}_f, \quad \hat{Q}_f = M_{\hat{g}} V_f. \tag{5.12}$$

5.5 EXPERIMENTS

In this section we present qualitative and quantitative results of our proposed framework on the Hot-or-Not and the MovieLens dataset.

5.5.1 *Hot-or-Not*

The dataset

Our dataset was kindly provided by Blinq[2], a popular hot-or-not dating application. We will make the anonymized ratings and visual features of the last layer available under `http://www.vision.ee.ethz.ch/~rrothe/`. The app shows the user people of the sex of interest, one after the other. The user can then like or dislike them. If both like each other's profile photo they are *matched* and can chat to each other. People can select up to 5 photos from Facebook for their profile.

Dataset statistics. Before performing any experiments we removed underage people, anyone over 37 and bi- and homosexual users as these comprise only a small minority of the dataset. All users who received less than 10 ratings were also removed. As the majority of the people decide on the first photo, we ignore the other photos. The resulting dataset has 4,650 female users and 8,560 male users. The median age is 25. In total there are 17.33 million ratings, 11.27m by men and 6.05m by women. Interestingly, 44.81% of the male ratings are positive, while only 8.58% of the female ratings are positive. Due to this strong bias of ratings by women, we only predict the ratings of men. There are 332,730 matches.

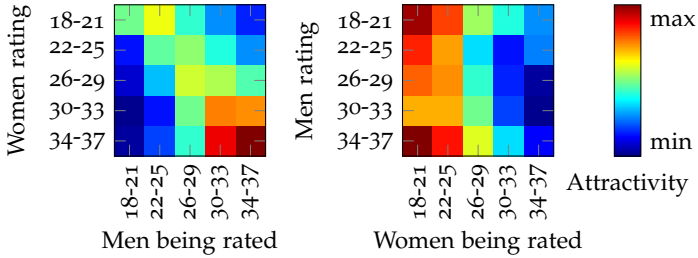


Figure 5.5: Preferences by age for women and men.

Preferences bias. To investigate the natural bias caused by age, we divide the men and women from our dataset according to their age

[2] www.blinq.ch

and gender. For each age group of men we counted the percent of hot vs. not on the ratings towards the age groups of women and vice versa. Fig. 5.5 describes the preferences by age as found in our dataset. Women generally prefer slightly older men and give better ratings the older they get. In comparison, men on this app prefer women under 25.

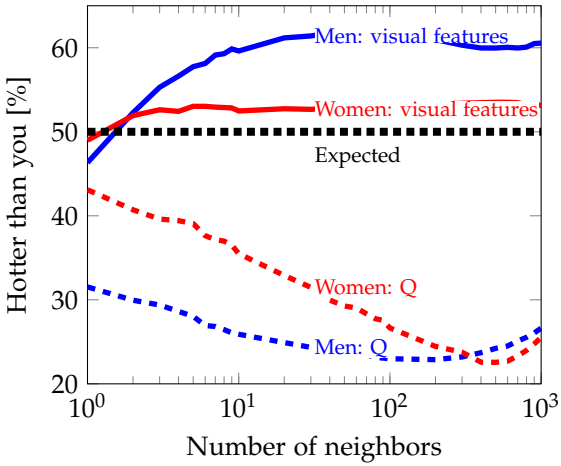


Figure 5.6: Hotness paradox. The people visually similar to you are on average hotter than you. The situation changes when we compute the similarity based on learned latent Q representations.

Hotness paradox. We notice an interesting phenomenon. Most people have a lower rating than their visually similar neighbors, on average, and this holds for both men and women. In Fig. 5.6 we report the percentage of cases where the average hotness of the neighborhood of a subject is greater than that of the subject, and this for a neighborhood size from 1 to 10^3. We plot also the results when we use our latent Q representations for retrieving neighbors. Surprisingly, this time the situation is reversed, the subjects tend to be hotter than their Q-similar neighborhood. Regardless the choice of similarity we have a strong deviation from the expected value of 50%. We call this phenomenon the 'Hotness paradox'. It relates to

the so-called 'Friendship paradox' [41] in social networks, where most people have fewer friends than their friends have.

Visual features. As the space covered by the person varies greatly between images, we run a top face detector [103] on each image. Then we crop the image to the best scoring face detection and include its surrounding (100% of the width to each side and 50% of the height above and 300% below), to capture the upper-body and some of the background. If the face is too small or the detection score too low, we take the entire image. Then we extract CNN features.

Experimental setup

For all experiments, 50% of either gender are used for training and the rest for testing. For each user in the testing set, 50% of the received ratings are used for testing. We compare different methods. *Baseline* predicts the majority rating in the training set. Matrix factorization is applied without and with visual regularization, *MF* ($\alpha_1 = 0.1$) and *MF+VisReg* ($\alpha_2 = 0.1$), resp. The dimensionality of the latent vector of P and Q is fixed to 20. The other parameters were set through cross-validation on a subset of the training data. We predict the ratings a subject receives based upon different knowledge: For *Visual* we solely rely on the image of the subject which means that we do not know any ratings the subject has received so far. For *10 Ratings, 100 Ratings, Full History*, we instead base the prediction upon a fixed set of known ratings for each query user. We report the average accuracy, *i.e.* the percentage of correctly predicted ratings of the testing set, and the Pearson's correlation.

Results

Performance. Fig. 5.7 shows how the average accuracy varies with the number of known past ratings for the query user. We report the average performance across all men's ratings. Knowing just the image of the person, we can predict 75.92% of the ratings correctly. Adding past received ratings of the user improves performance to up to 83.64%. Matrix factorization significantly improves as more ratings are known. If only few ratings are available, regularizing

the matrix factorization with the visuals boosts performance significantly, *i.e.* from 72.92% to 78.68% for 10 known ratings. Table 5.3 summarizes the results for various settings.

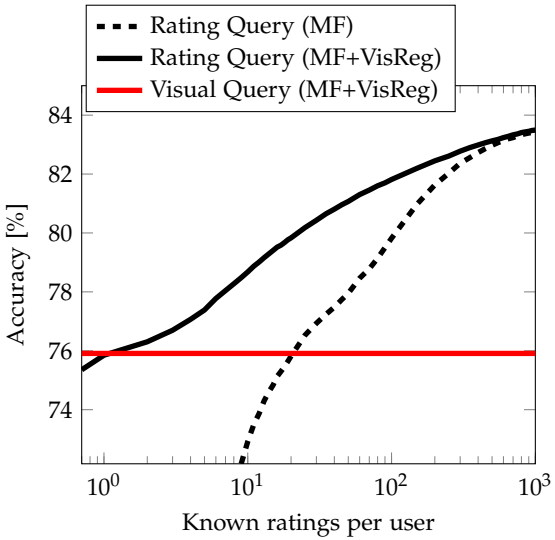


Figure 5.7: Number of known ratings for a female query user vs. accuracy of predicted male's ratings.

Table 5.3: Preference prediction results on Hot-or-Not dataset for female queries.

	Query	Accuracy	Correlation
Baseline	N/A	54.92%	N/A
MF	Visual	75.90%	0.520
MF+VisReg		**75.92%**	**0.522**
MF	10 Ratings	72.92%	0.456
MF+VisReg		78.68%	0.576
MF	100 Ratings	79.82%	0.593
MF+VisReg		81.82%	0.635
MF	Full History	83.62%	0.671
MF+VisReg		**83.64%**	**0.671**

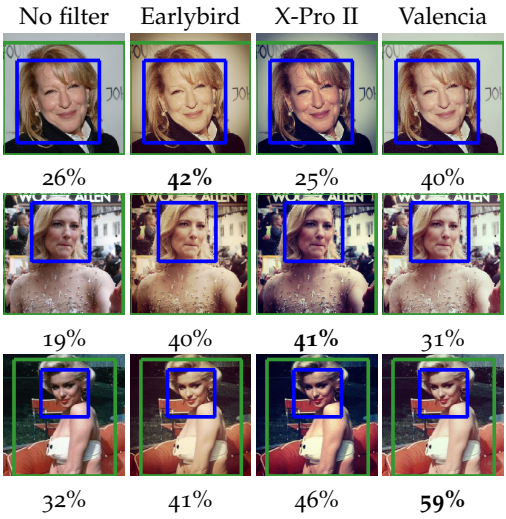


Figure 5.8: Improving the hotness rating by Instagram filters.

Latent space Q vs. preferences. In Fig. 5.9 we show the learned latent space Q from the matrix factorization by PCA projecting it to two dimensions and adding the hotness and age properties for both genders with visual regularization. The learned latent factor Q captures appearance and for women there is a clear separation in terms of attractiveness and age, whereas for men the separation is less obvious.

According to the preferences P and the learned latent Q one can have a more in-depth view on the appearance of women and men. In Fig. 5.10 both men and women are clustered according to their 2D representation of the learned latent factors P (preferences of men) and Q (appearances of women), respectively. For visualization purposes we used 100 user images for each cluster and 10 clusters. The men are visually more diverse in each of their clusters than the women in their clusters, because the men are clustered according to their preferences, therefore ignoring their visual appearance, while the women are clustered according to their Q factors which are strongly correlated with appearance and hotness, as shown in Fig. 5.9.

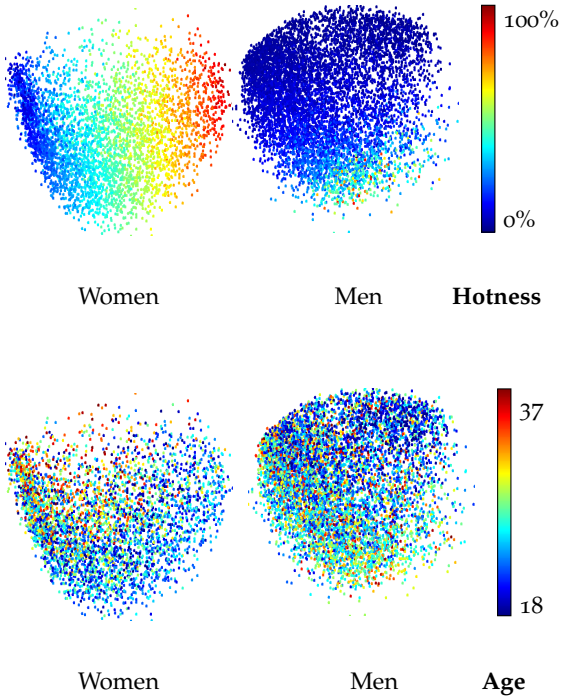


Figure 5.9: Visualization of latent space Q for women and men.

Visual queries without past ratings. We validate our approach on images outside our dataset, retrieved from the Internet for celebrities. By applying the visual query regression to the Q space we can make good predictions for such images. For a visual assessment see Fig. 5.12. This figure also depicts a number of issues our pipeline faces: too small faces, detector failure, wrongly picked face, or simply a wrong prediction. We also tested our method on cartoons and companion pets with the predictor trained on Hot-or-Not. The results are surprising.

Instagram filters or how to sell your image. Images and their hotness prediction also indicate which changes could improve their ratings. Earlier work has aimed at the beautification of a face image by invasive techniques such as physiognomy changes [93] or makeup [95], [105]. Yet, we found that non-invasive techniques

Figure 5.10: Preferences between clusters of users. The color of the arrow indicates how much the men's cluster likes (green) or dislikes (red) the women's cluster on average.

(not altering facial geometry and thus 'fair') can lead to surprising improvements. We have evaluated the most popular Instagram filters[3] for our task. We observed that the filters lead to an increase in predicted hotness. In Fig. 5.8 we show a couple of results in comparison to the original image. Note that with our predictor and such Instagram filters a user can easily pick its best profile photo.

howhot.io

We demonstrate our algorithms on `howhot.io`, a website where people can upload a photo of their face and our algorithm will then estimate the age, gender and facial attractiveness of the person (c.f. Fig. 5.11). The CNN was trained on the Hot-or-Not dataset for predicting attractiveness and on the IMDB-WIKI dataset [125] for age and gender prediction. The website went viral around the Internet with more than 50 million pictures evaluated in the first month.

[3] brandongaille.com/10-most-popular-instagram-photo-filters

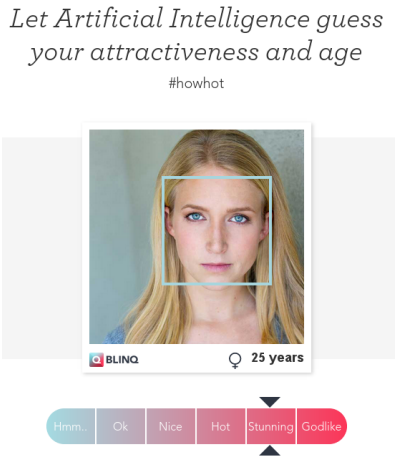


Figure 5.11: howhot.io – a website utilising deep learning to predict age, gender and attractiveness of a person. More than 1 million users visited the website in the first 12 hours and more than 50 million photos were uploaded in the first month.

5.5.2 *MovieLens*

The dataset

We also perform experiments on the MovieLens 10M[4] dataset. It contains 10,000,054 ratings from 69,878 users for 10,681 movies. Ratings are made on a 5-star scale, with half-star increments. On average, each movie has 3,384 ratings and each user rates 517 movies. Note that even though there are more than 10 million ratings, the rating matrix is sparse with only 1.34% of all ratings known. We augment each movie with the poster image from IMDB and extract the same deep CNN features as for the Hot-or-Not dataset. We will make the poster images publicly available under `http://www.vision.ee.ethz.ch/~rrothe/`.

[4] grouplens.org/datasets/movielens

Table 5.4: Rating prediction results on augmented MovieLens.

	Query	MAE	Correlation
Baseline	N/A	1.507	N/A
MF	Visual	0.824	0.286
MF+VisReg		**0.813**	**0.292**
MF	10 Ratings	1.031	0.280
MF+VisReg		0.872	0.270
MF	100 Ratings	0.740	0.467
MF+VisReg		0.780	0.461
MF	Full History	0.696	0.530
MF+VisReg		**0.696**	**0.536**

Experimental Setup

The experimental setup in term of training and testing split is identical to the Hot-or-Not dataset. As the movie posters are much less informative regarding the ratings in comparison to the Hot-or-Not images, the visual regularization is reduced to $\alpha_2 = 0.001$. For a given movie we want to infer the ratings of all users. Again, we evaluate the case where just the poster is known and also cases where a varying number of ratings is known. As a baseline we show how a prediction made at random would perform, assuming that there is no bias in the ratings of the test set.

Results

Table 5.4 summarizes the performance. Fig. 5.14 shows how the number of known ratings impacts the MAE. Visual regularization of MF improves performance, especially when few ratings are known, *i.e.* for 10 known ratings the MAE can be reduced by 15% from 1.031 to 0.872. When just the movie poster is known, the MAE is 0.813, which is on par with knowing 30 ratings. Fig. 5.13 shows some movie posters. We also show overrated and underrated posters, *i.e.* posters where our algorithm - based on the poster - predicts a much better or worse score than the actual movie rating.

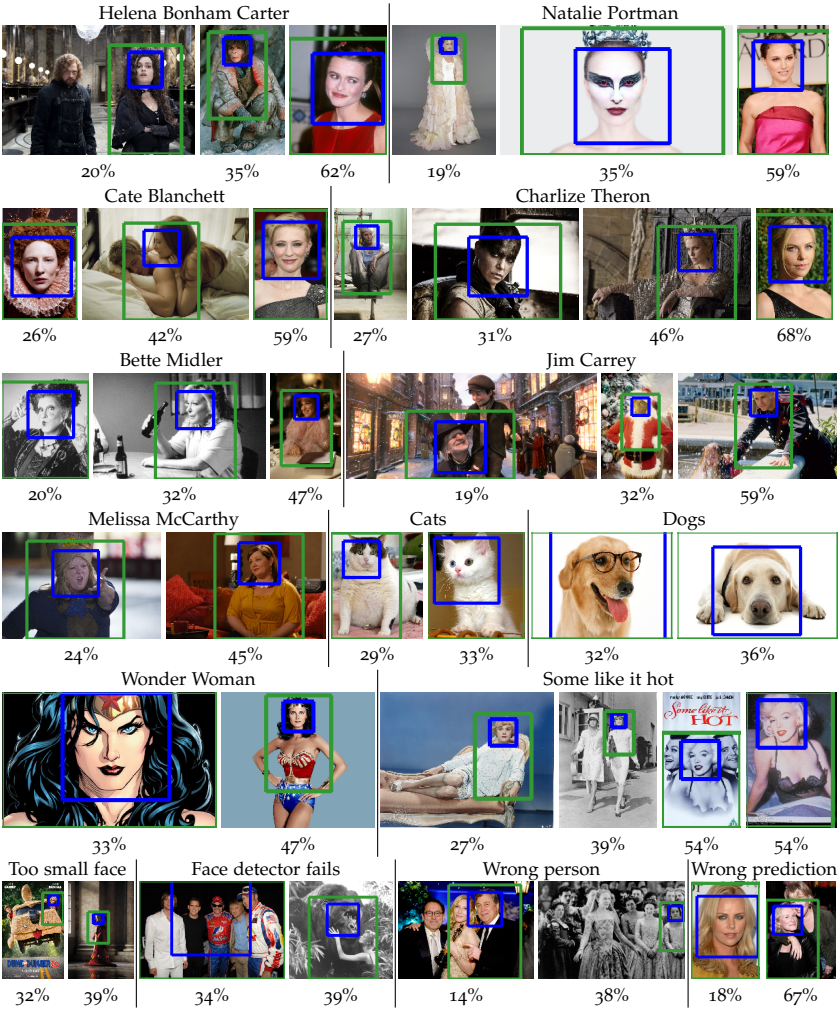


Figure 5.12: Predicted percentage of positive ratings for numerous celebrities by the user base of the Hot-or-Not dataset.

5.6 CONCLUSION

We proposed a collaborative filtering method for rating/preference prediction based not only on the rating history but also on the

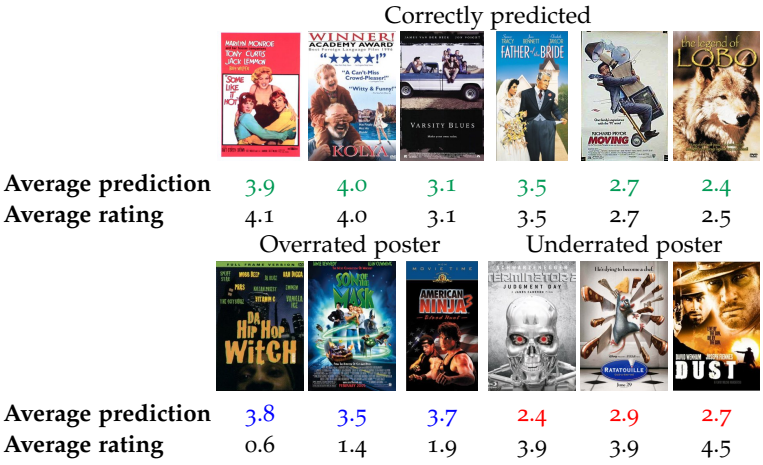


Figure 5.13: Examples of predicted ratings for various movie posters solely based on the visual information of the poster.

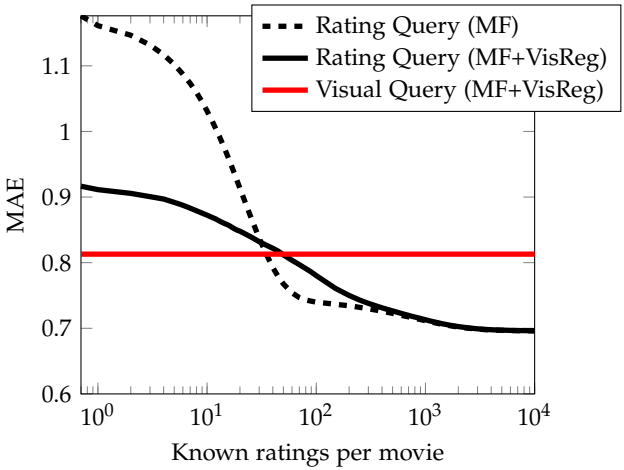


Figure 5.14: Number of known ratings for a movie vs. MAE of the predicted ratings.

visual information. Moreover, we can accurately handle queries with short or lacking rating history. We evaluated our system on a very large dating dataset and on the MovieLens dataset augmented with poster images. To the best of our knowledge, we are the first to report on such a large dating dataset, and to show that adding weak visual information improves the rating prediction of collaborative filtering methods on MovieLens. We achieved state-of-the-art results on facial beauty, age and gender prediction and give some sociologically interesting insights. Thus this chapter can be seen as an extension of the work in Chapter 4 by providing a fine-grained classification framework which is applicable to subjective rather than objective facial attributes. In the next chapter we go beyond fine-grained classification for face images and present a pipeline for event classification.

6 DEEP RETRIEVAL FOR CULTURAL EVENT CLASSIFICATION

6.1 INTRODUCTION

Following the fine-grained classification applications in the previous two chapters, in this chapter the focus is on the fine-grained classification of cultural events. Image classification is at the core of computer vision. Extensive literature is devoted to the study of classification of images into objects and/or scenes. The recent advances due to the introduction of large datasets such as ImageNet [129] or PASCAL VOC [39] for object classification and the use of deep learning techniques [23], [52], [77] brought into focus the classification of more demanding classes such as 'cultural events' where the geometry and/or appearance of a single object or scene are not anymore the dominant features determining the class. Particularly, a picture of a cultural event depends entirely on the photographer's subjectivity. Each such picture is just a narrow view of what happens under the big umbrella of the cultural event. Classification and retrieval of images of cultural events are of interest for many people, especially tourists. There are multiple important cultural events in the world that attract lots of participants and produce huge amounts of photos to browse.

In this chapter we tackle the classification of cultural events from a single image, a consumer photograph, with a deep learning-based method and report our performance on the cultural event recognition dataset of the ChaLearn Looking at People 2015 (LAP) challenge [38] (see Fig. 6.1).

We use convolutional neural networks (CNNs) with VGG-16 architecture [134], pretrained on the ImageNet dataset [129] or the Places205 dataset [183] for image classification, and fine-tuned on cultural events training data from LAP. Our CNN features are the fully-connected (fc) layer 7 with 4096 dimensions. We follow a layered approach (see Fig. 6.2). For each layer, CNN features are

Figure 6.1: Cultural event images and class labels from LAP dataset.

robustly extracted from each image over a grid. At each layer, Linear Discriminant Analysis (LDA) [45] is employed for reducing the dimensionality of the CNN features and to embed discriminativity. An image is represented by the concatenated LDA-projected features from all layers or by the concatenation of the average pooled raw CNN features at each layer. The classification is handled through the Iterative Nearest Neighbors-based Classifier (INNC) [145], [146]. Classification scores are obtained for different image representation setups at train and test. The average of the scores is the output of our deep linear discriminative retrieval (DLDR) system. DLDR is a top entry for the ChaLearn LAP 2015 cultural event recognition challenge with 0.80 mean average precision (mAP), 0.05 below the best reported result.

Next we review work related to our task and method. Section 6.2 introduces our DLDR method. Section 6.3 describes the experiments and discusses the results, while in Section 6.4 we conclude the chapter.

6.1.1 *Related work*

The ChaLearn Looking at People challenge on cultural event recognition from single images in conjunction with CVPR 2015 [6] is the precursor of the ChaLearn LAP challenge in conjunction with ICCV 2015 [38] that we targeted in this chapter. The previous challenge used a 50 classes dataset while the new one extended it by proposing a larger dataset with 100 classes. The solutions proposed for the previous challenge are those most related to our own. In Table 6.5 are the top 4 ranked teams of that challenge. Next, we present them in relation to our proposed DLDR method.

MMLAB: The solution of Wang *et al*. [158] fuses five types of CNNs. These are ClarifaiNet [179] pretrained on the ImageNet dataset, AlexNet [77] pretrained on the Places205 dataset, GoogleNet [137] pretrained on the ImageNet dataset and the Places205 dataset, and VGG-19 [134] pretrained on the ImageNet dataset. All of them are fine-tuned on the cultural event training data and the scores are fused by weighting for the final results. MMLAB ranked 1^{st} with 0.85 mAP, significantly more than the next team with 0.76 mAP. Our DLDR is significantly lighter, it uses only one kind of CNN, the VGG-16, pretrained on ImageNet and on Places205. DLDR also fine-tunes and fuses scores for the final results, but in addition uses multiple layers in the representations, discriminant projections, and INNC classifiers.

UPC-STP: The team of Salvador *et al*. [130] combines features from the fully connected (fc) layers of a CNN pretrained with ImageNet and a second one fine-tuned on the cultural event dataset. For each fc layer, Linear SVMs are trained for the corresponding features. These are further fused using an SVM. A temporal model of the events is learned and used to refine the outputs. Our DLDR uses another CNN architecture, pretrains also on ImageNet, uses only the last fc layer as CNN raw features and employs a different classification strategy.

MIPAL_SNU: The team of Park and Kwak [112] assumes that the discriminant image regions are the ones relevant to classification. Therefore, they first extract meaningful image regions of various size. Then they train a CNN with 3 convolutional layers and pooling layers, and 2 fc layers. The probability distribution for the

classes is calculated for every image region selected from the test image and class probabilities are computed.

SBU_CS: The team of Kwon *et al*. [80] studies SIFT, SIFT+color, and CNN features in combination with 3 layer spatial pyramid matching (SPM) [87] and a regularized max pooling (RMP) [67] technique. The CNN is pretrained on ImageNet and no fine-tuning is employed. Their best combination is a SPM with SIFT+Color and RMP with CNN features. Our DLDR method also uses layered representations and CNN features.

The novelty of our proposed method lies in using LDA discriminative projections of CNN features at different pyramidal layers and per layer pooled CNN features to improve classification accuracy. Furthermore, we extend the formulation of the INNC classifier with weight-spreading to better deal with retrieval of a large number of classes.

6.2 PROPOSED METHOD (DLDR)

In this section we describe the proposed method: deep linear discriminative retrieval (DLDR). The scheme of DLDR is shown in Fig. 6.2.

6.2.1 *Deep learning*

Our DLDR is based on the deep learned representations of image regions. We employ CNNs with the VGG-16 [134] architecture which provides a good balance between the representation power and time plus memory requirements. Simonyan *et al*. [134] achieve state-of-the-art results with this architecture on benchmarks such as ImageNet [129].

Pretraining

Without a (very) large training set of images getting trained from scratch a CNN with a very large number of parameters (like ours) is cumbersome and likely to overfit and to produce poor results. Therefore, for cultural events recognition we use as a starting point the CNN pretrained on the ImageNet dataset [134] and CNN pre-

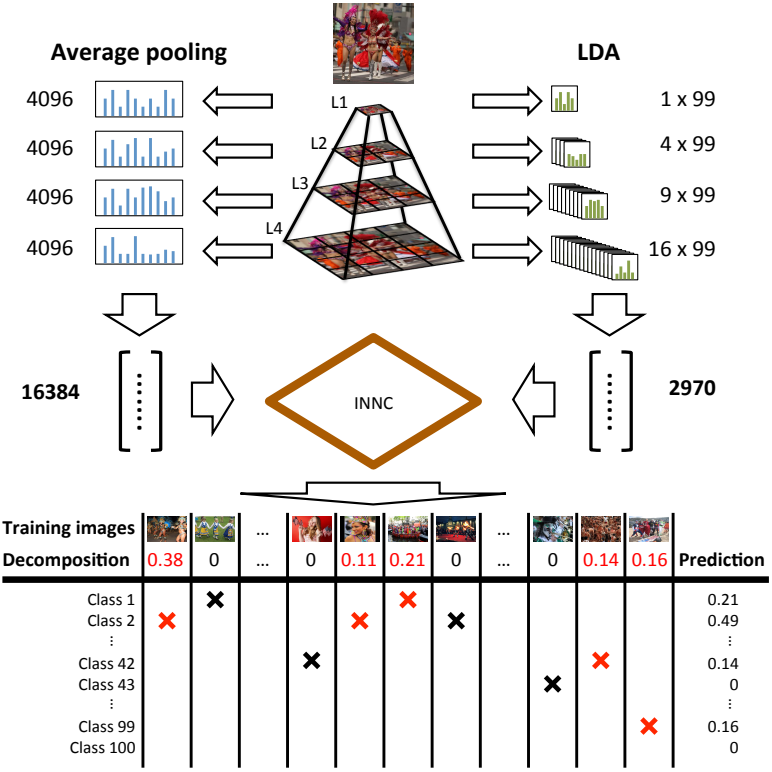


Figure 6.2: Pipeline for our DLDR method.

trained on the Places205 dataset [157]. Previously Wang *et al*. [158] also used these two datasets for pretraining models for cultural event recognition.

Training

We train two separate CNNs on the provided LAP dataset corresponding to those pretrained on ImageNet and Places205, resp. We adapt the output layer of the network to have a number of neurons equal to the number of classes, here 99 cultural events and a 'Non-Class' as in the LAP dataset. The training data, consisting of the

provided LAP training dataset and LAP validation dataset, was randomly split into 90% used for training and 10% for testing. We kept the distribution of classes the same in both sets. Our training set is further enlarged by augmentation. 10 random crops from each original training image are added to the training set. Each random crop has at least half the side length of the original image.

6.2.2 *Layered representations*

Inspired by [66], [87], we extract CNN features in a pyramidal fashion. Specifically we extract features at 4 scales. In the first level we extract features over the entire image, in the second, third, and fourth level we extract from 2×2, 3×3, and 4×4 regions, resp. The regions overlap with 50%. We scale each image region to 256×256 and then extract the last feature layer (fc7, 4096 dimensions) for 10 different crops at a size of 224×224 in each corner and the center of the image, as in [52]. We do the same for the flipped version of the image. The features of these 10 crops are then averaged to give the final feature representation. This results in $1^2 + 2^2 + 3^2 + 4^2 = 30$ feature representations of 4096 dimensions.

We can not handle representations of 30 concatenated raw CNN features. We either employ encoding over a visual codebook as in standard SPMs (we got discouraging preliminary results), reduce the dimensionality (i.e. through LDA), or pool the raw features at each layer.

Pooled CNN features

The idea of pooling directly the raw CNN features without caring about their image positions is inspired by the robust prediction commonly employed by CNN solutions (predicting on different crops around the desired image region of interest). We considered different pooling operators and found average pooling to be the best in robustness and performance. Our pooled features are average pooled raw CNN features at a given layer. Correspondingly, the layered representation, called R1, has $n \times 4096$ dimensions where n is the number of layers with pooled raw CNN features. In

our case the representation is high dimensional ($4 \times 4096 = 16384$ dimensions) and thus can capture subtle details learned by the CNN.

LDA-projected features

Due to limited computational resources we explored efficient dimensionality reduction methods. Principal component analysis (PCA), a natural choice, loses quite a bit of performance even for reduction factors of 2 or 4. Since reducing the dimensionality while preserving the energy is challenging, we picked linear discriminant projections that would thus compensate for the loss in dimensions by improving the discriminative power.

Linear Discriminant Analysis (LDA) maximizes the ratio of the between-class scatter and the within-class scatter. We use LDA in its regularized form [45], with regularization parameter set to 1, as implemented by Cai *et al*. [13]. In our preliminary experiments, LDA and its 99-dimensional projections (number of classes - 1) were able to provide for equal and better classification performance than the original raw features, while SRLP [147] (which embeds sparse relations) needs 200 dimensions to improve over the LDA-projections.

We learn a separate Linear Discriminant Analysis (LDA) projection for each of the 4 layers in our representation. We then concatenate the LDA-projected features to form a feature vector of $30 \times 99 = 2970$ dimensions, representation R2. Additionally we construct a flipped representation of R2 by horizontally flipping the local representation for the 2^{nd}, 3^{rd} and 4^{th} layer, called R3. Note that R3 is a permutation of the features of R2.

The LDA helps to not only reduce the dimensionality but also to embed discriminativeness into the features.

6.2.3 *Classification*

For classification we use the Iterative Nearest Neighbors-based Classifier (INNC) of Timofte and Van Gool [146]. The INN representation [145] is the result of a sparse linear decomposition of the query sample over the training pool. The weights belong to

Table 6.1: mAP (%) on our validation set (2863 of 20036 images) for different configurations.

Layers	Encoding	CNN pretrained on ImageNet	CNN pretrained on Places205	Fusion
L1	Raw	74.59	73.61	77.32
	LDA	73.91	73.96	77.64
L2	Raw	76.16	73.70	77.90
	LDA	77.90	75.92	79.69
L3	Raw	75.43	72.20	76.54
	LDA	77.65	75.39	79.03
L4	Raw	73.63	69.00	74.18
	LDA	77.28	73.14	77.73
L1+L2	Raw	76.75	75.16	78.75
	LDA	78.00	76.62	80.05
L1+L2+L3	Raw	77.52	75.80	79.24
	LDA	79.00	77.12	80.22
L1+L2+L3+L4	Raw	77.63	75.84	79.25
	LDA	79.10	76.93	80.12

Table 6.2: mAP (%) of DLDR on our validation set (2863 of 20036 images).

Train/test representation	ImageNet	Places205	Fusion
C1	77.63	75.84	79.25
C2	79.10	76.93	80.12
C3	79.26	76.77	80.16
C4	79.36	77.08	80.38
C2+C3+C4	79.61	77.29	80.47
C1+C2	79.56	77.56	80.46
C1+C2+C3+C4	**79.96**	**77.74**	**80.70**

Table 6.3: Classification on our validation set (2863 of 20036 images)

	ImageNet	Places205	Fusion
Linear SVM	77.04	75.58	78.87
INNC	78.42	76.15	79.76
INNC-KNN	79.10	76.93	80.12

$[0, 1)$ and sum up to 1. For each class, the weights corresponding to training samples of that class are summed up. The class with the largest impact in the INN decomposition of a query is the INNC prediction. We set the maximum number of non-zeros (neighbors) to $K = 14$ and the regularization parameter to $\lambda = 0.1$. For each test sample we obtain an INN representation over the training set. This sparse matrix of weights is then used for classification. The probability for a given test sample to belong to a class is taken as the sum of the weights corresponding to all training samples of that class. As the INN representation is sparse ($\leq K$), often with fewer non-zero weights than classes, many classes have a probability of 0. To overcome this issue we extend the formulation of INNC by additionally spreading the weights also to the nearest neighbors of the training samples, with some exponential decay (0.75^r, where r is the rank of the neighbor). This helps to increase retrieval performance especially on difficult samples.

INNC is applied to the representations separately:

C1: R1

C2: R2 in the training set and R2 at testing

C3: R3 in the training set and R2 at testing

C4: R2 and R3 in the training set and R2 at testing

Note that if we would have had R2 in the training set and R3 at testing, this would be the same as C3 as R2 and R3 just differ by permutation. We obtain those predictions for both networks, resulting in 8 predictions in total which are averaged fused to give the final DLDR prediction score.

6.3 EXPERIMENTS

6.3.1 *Dataset and evaluation protocol*

The ChaLearn LAP cultural event recognition dataset [38] consists of 28705 images collected from two images search engines (Google Images and Bing Images). The images contain photos from 99 important cultural events around the world and a non-class. The

dataset is split into three parts, 50% for training (14332 images), 20% for validation (5704 images), 30% for testing (8669 images). There are approximately the same number of images in each class with the exceptions of the non-class having around ten times as many images.

In this chapter the results are evaluated as defined for the ChaLearn LAP challenge. Specifically, for a given class the average precision (AP) is calculated by measuring the area under the precision/recall curve. The AP scores are then averaged over all 100 classes to form the final mean average precision (mAP).

6.3.2 *Implementation details*

Our DLDR pipeline is written in Matlab. The CNNs are trained on Nvidia Tesla K40C GPUs using the Caffe framework [72]. The machine used for calculating the LDA projections and classification has 128 GB of memory.

Training each of the two CNNs took about 30 hours. At test time extracting the features over all training and testing images over all 4 layers ($30 \times 10 = 300$ extractions per image) took around 100 hours. Calculating the LDA projections and classification took around 3 hours.

The source codes are publicly available at:
`http://www.vision.ee.ethz.ch/~timofter`

6.3.3 *Validation results*

We compare the performance of our proposed method for different feature representations, layers and pretrained networks. In Table 6.1 the results are summarized. The performance is shown for the case where features are extracted only at one layer, as well as for the case when features with increasing depth are combined. For both the pooled CNN features as well as the LDA-encoded features the performance increases from L1 to L2. Beyond L2, for L3 and L4 performance decreases again suggesting that the level of detail gained from the smaller image regions cannot compensate the insights missing from a global feature representation.

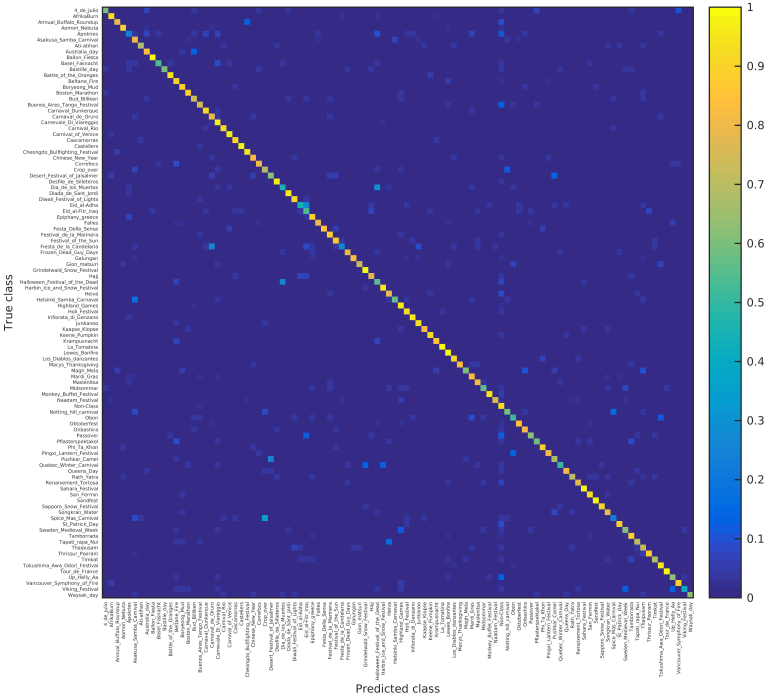


Figure 6.3: Confusion matrix for our DLDR system on the LAP classes. Best seen on screen.

Nonetheless, when combining the features from various layers, performance increases as one adds more layers to the representation. This suggests that the increase in dimensionality of the feature vector is well compensated by the extra insights from the smaller image regions. Comparing the pooled CNN features with the LDA-projected features, in the majority of the experimental setups, the LDA features perform around 2% better. The weights after pretraining on ImageNet give consistently better results than the Places205 weights. However, the fusion of the two then again improves performance by at least 1%. Overall we are able to push the performance from 74.59% to 80.12% when combining the LDA-projected features from all layers and the ImageNet and the Places205 pretraining compared to when just extracting pooled features from ImageNet on the entire image (L1).

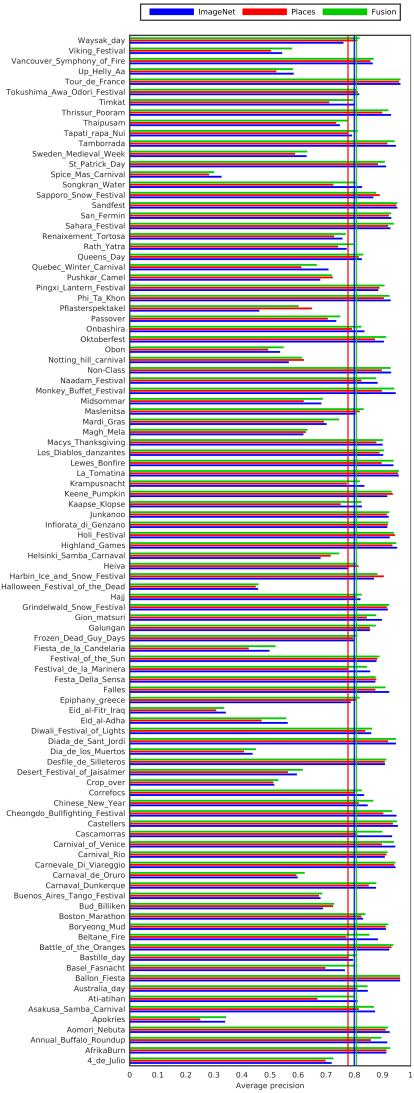


Figure 6.4: DLDR average precisions (AP) for LAP classes using Places205 pretraining, ImageNet pretraining, or the fused predictions.

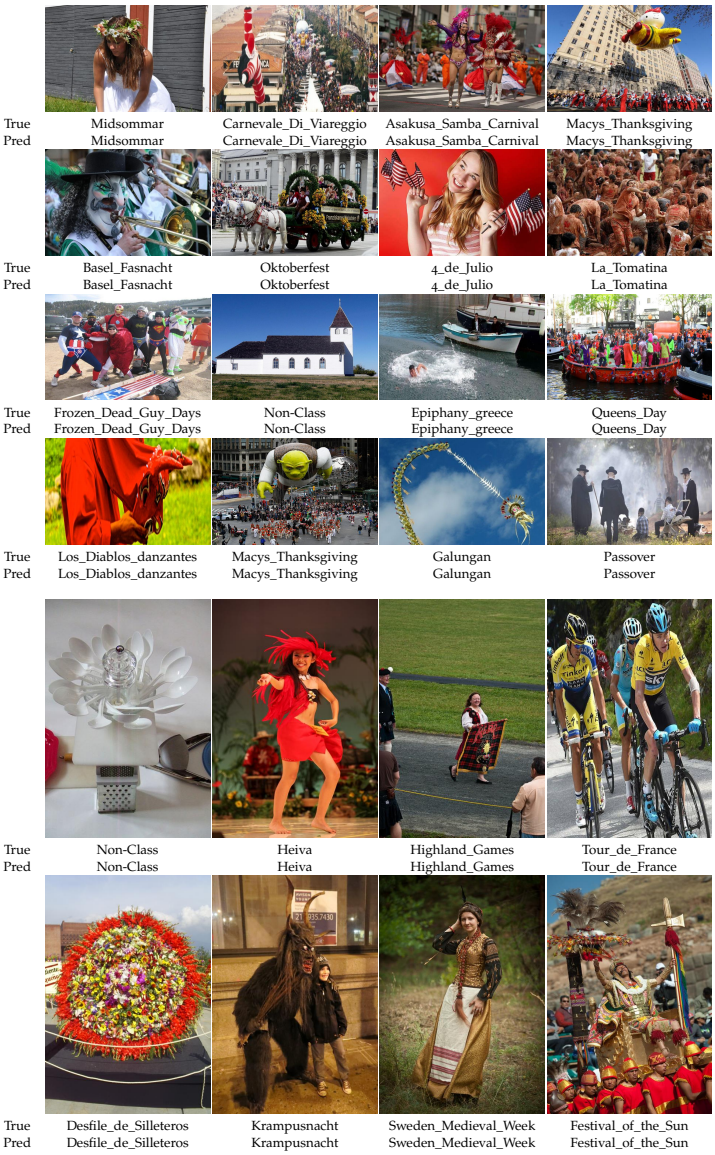


Figure 6.5: Examples of images where DLDR is successful in a top-1 evaluation.

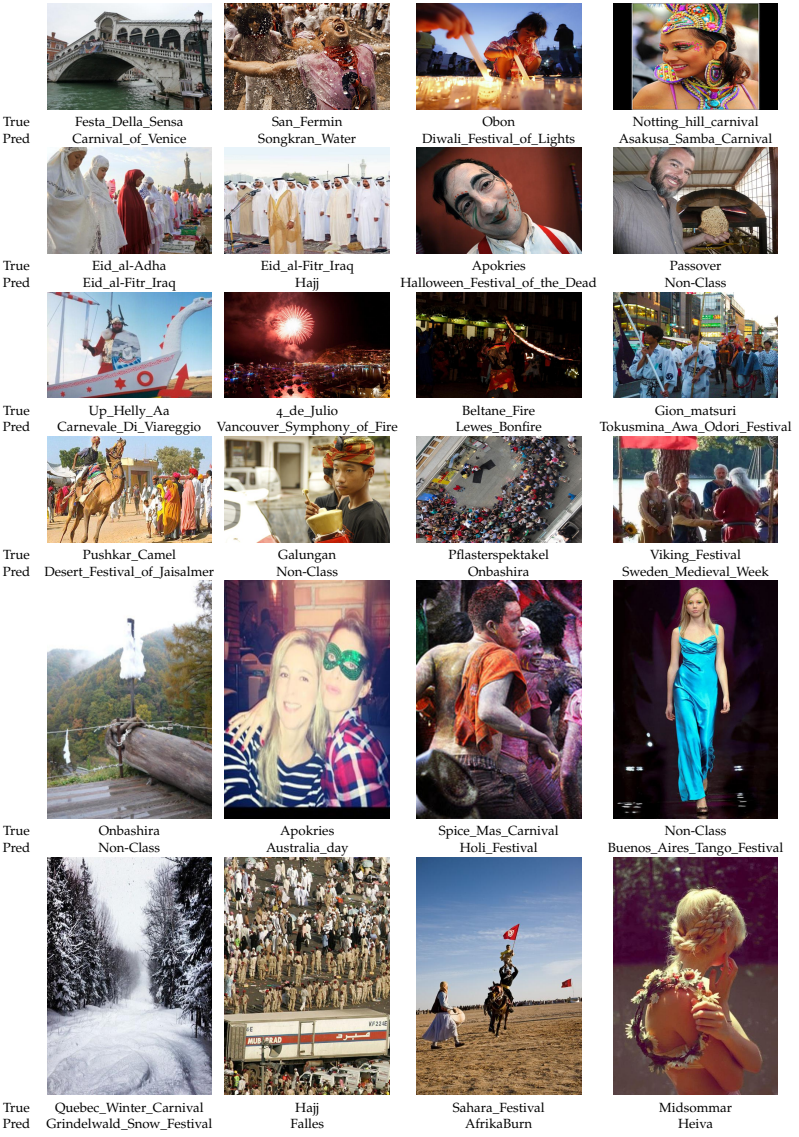


Figure 6.6: Examples of images where DLDR fails in a top-1 evaluation.

For classification we compare the performance of our proposed INNC with weight spreading (INNC-KNN) to the conventional INNC and Linear SVM as shown in Table 6.3. When fusing the features from ImageNet and the Places205 dataset INNC improves 1% over Linear SVM. Weight spreading further improves performance by 0.5%

Table 6.2 shows the performance when combining the 4 different classifications for each network, resulting in 8 predictions in total. Combining the LDA-projected predictions with its flipped version (C2+C3+C4) improves performance by around 0.5%. Also combining the LDA-projected features with the pooled CNN features gives an improvement of 0.5% over the LDA features and more than 1% over the pooled features. Combining all 8 predictions then leads to an overall improvement of 1.5% over just using the pooled CNN features. Overall this improves the performance of just using R1 by 1.5% up to 80.70% on our validation set.

In Figure 6.4 we compare the performance of the pretrained ImageNet network and the Places205 network. For the majority of the classes the fusion of the two networks outperforms the individual networks. Pretraining on ImageNet generally gives better results than when the network was pretrained on the Places205 dataset. However, there are some exceptions, i.e. for the classes Pflasterspektakel, Waysak_day, and Pushkar_Camel the pretrained Places205 network seems to give better accuracy.

As some of the classes are very similar, i.e. there are multiple carnival events, we investigated the confusion between classes. Specifically we assigned each image to the class with the largest confidence and then visualized the inter-class confusion (see Fig. 6.3). Some classes like Eid_al-Adha and Eid_al-Fitr_Iraq or Pushkar_Camel and Desert_Festival_of_Jaisalmer have a high confusion, which is also confirmed when looking at the images from the classes – as a human it is nearly impossible to distinguish between them.

In Figure 6.5 we visualize cases where our proposed method successfully recognizes the correct class. The system seems to successfully pick up subtle details which are typical for the event (i.e. the US flag for 4th of July or the floral wreath for Midsommar).

Figure 6.6 shows some failure cases. In many of those cases the classes are either very similar (i.e. same type of event, same location, same vegetation) or the image shows just one large object and it is thus hard to directly assign it to a specific class (i.e. just a person, a boat or a building).

6.3.4 *Looking At People (LAP) challenge*

The ChaLearn Looking at people (LAP) challenge on cultural event recognition had two phases.

In the first phase, the training and validation images of the LAP dataset were provided to the registered participants. If the training images had class labels, the labels for validation images were unknown until the second phase. For the performance score (mAP) on the validation set each team submitted their results to the server. After the validation phase, the labels for the validation images were released together with the test images. Again, the teams were invited to submit their results on the test images to the competition server without getting to know their performance or rank. The organizers announced the final ranking and scores after the second phase ended. Table 6.4 shows the final ranking of the ChaLearn LAP challenge on cultural event recognition based on the test set. Our DLDR method ranks 5^{th} with a mAP of 0.80, being only 0.05 below the best reported performance of the VIPL-ICT-CAS team.

This ChaLearn LAP challenge with 100 classes was preceded by a ChaLearn LAP challenge in conjunction with CVPR 2015 which had 50 classes [6]. Most of the top ranked teams, unlike us, participated also in the previous challenge. The top 4 teams in the previous (easier) challenge are listed in Table 6.5 and their solutions were discussed in the related work section 6.1.1.

Table 6.4: ChaLearn LAP 2015 final ranking on the test set. 67 registered participants.

Rank	Team	mAP
1	VIPL-ICT-CAS	0.85
2	FV	0.85
3	MMLAB	0.84
4	NU&C	0.82
5	**CVL_ETHZ (ours)**	**0.80**
6	SSTK	0.77
7	MIPAL_SNU	0.76
8	ESB	0.76
9	Sungbin Choi	0.62
10	UPC-STP	0.58

Table 6.5: CVPR ChaLearn LAP 2015 top 4 ranked teams [6]

Rank	Team	mAP
1	MMLAB [158]	0.85
2	UPC-STP [130]	0.76
3	MIPAL_SNU [112]	0.73
4	SBU_CS [80]	0.61

6.4 CONCLUSIONS

We proposed an effective method for cultural event recognition from single images called Deep Linear Discriminative Retrieval (DLDR). DLDR employs CNNs pretrained on ImageNet and Places205 datasets, and fine-tuned on cultural events data. CNN features are robustly extracted at four different layers in each image. They are either average pooled or LDA projected at each layer. Thus, an image is represented by the concatenated LDA-projected features from all layers or by the concatenation of CNN pooled features at each layer. Using our Iterative Nearest Neighbors-based Classifier (INNC), scores are obtained for different image representation setups. The average scores are the fused DLDR output. With 0.80 mean average precision (mAP) our DLDR solution is

a top entry in the ChaLearn LAP 2015 cultural event recognition challenge.

7 CONCLUSION

How much can a computer infer from a single image? In this thesis, we aimed at answering this question by proposing various techniques including image processing, object detection, and fine-grained classification. For all those methods, the inference is based on just one image. In this chapter, we will summarize the contributions of this thesis and give an outlook on potential future work.

7.1 SUMMARY

The main contributions of this thesis are provided in five different directions with the overarching goal to improve single-image understanding:

- In Chapter 2, we proposed an efficient novel artifact reduction algorithm based on the adjusted anchored neighborhood regression (A+) [141]. The proposed method doubles the relative gains in PSNR when compared to state-of-the-art methods such as Semi-local Gaussian Processes (SLGP) [82], while being order(s) of magnitude faster.

- In Chapter 3, we proposed a novel formulation of non-maximum suppression (NMS) as a post processing step for object detection for a single image. Our method is based on the recent Affinity Propagation Clustering algorithm [44] and contrary to the standard greedy approach solved globally with its parameters being learned automatically. The experiments showed for object class and generic object detection that it provides a promising solution to the shortcomings of the greedy NMS.

- In Chapter 4, we proposed a deep learning solution to age estimation from a single face image without the use of facial

landmarks. As part of this contribution we made the IMDB-WIKI dataset publicly available – the largest public dataset of face images with age and gender labels. Our method achieves state-of-the-art results for both real and apparent age estimation winning the Chalearn Looking at People (LAP) age estimation challenge [38] against 115 other competitors.

- In Chapter 5, we proposed a framework to infer visual preferences from profile images and user ratings. Our computational pipeline comprises a face detector, convolutional neural networks for the extraction of deep features, a novel visual regularized collaborative filtering to infer inter-person preferences as well as a novel regression technique for handling visual queries without rating history. We validated the method using a very large dataset from a dating site, images from celebrities as well as on the standard MovieLens rating dataset, augmented with movie posters.

- In Chapter 6, we proposed a framework for classifying cultural events from a single image. The method is based on extracting CNN features at multiple scales, which are then encoded using Linear Discriminant Analysis (LDA) and classified through the Iterative Nearest Neighbors-based classifier (INNC) [146]. The proposed method is a top entry for the ChaLearn LAP 2015 cultural event recognition challenge [38].

7.2 FUTURE WORK

Organized by the chapters of this thesis, in this section we elaborate on future directions of research in the context of the different applications.

Chapter 2, *Reducing image compression artifacts*

- **Video enhancement.** The proposed method achieved some impressive results for reducing image compression artifacts. With the rise of video data, this method could be extended to

be applied to videos. In that case, the regressors could then be learned both spatial and temporal.

- **Handling other corruptions.** So far, we have constrained our experiments to the reduction of compression artifacts. However, the same technique could also be applied in the case of other corruptions like blur. This might require to train the regressors specifically on images with these specific corruptions.

- **Context adaptive enhancement.** Timofte *et al*. [143] showed that context can help to improve the performance for super-resolution by learning class-specific regressors. Similarly for the reduction of image compression artifacts, a specific set of regressors could be learned for each class (*i.e*. people, cars, flowers).

Chapter 3, *Non-maximum suppression for object detection*

- **Detector and NMS jointly optimized.** In the current setup, the NMS procedure is seen as a post-processing step to an object detector. The beauty of this formulation is that it is independent of the detector and can be paired with any type of detection algorithm. The downside is that there is no joint optimization of the detector and the non-maximum suppression. Future work could seek to extend the current formulation to not only select the best bounding boxes to cover an object but also optimize the detector at the same time to give out more reasonable bounding boxes in the first place.

- **Deep Learning.** The experiments presented in Chapter 3 are all based on the detections of Felzenszwalb's DPM object detector [42]. Especially in recent years, deep learning has shown impressive results for object detection, *i.e*. the R-CNN framework [52], [117] achieves accuracy far beyond DPM. As this pipeline also employs a greedy NMS, it would be interesting to evaluate how much the performance can be pushed by our NMS procedure.

- **Context adaptive NMS.** The NMS procedure could be extended and potentially further improved by also considering context and class specific knowledge. Context could imply re-evaluating the image in the bounding box whereas class specific knowledge could be incorporated by training the NMS procedure separately for each class.

Chapter 4, *Predicting real and apparent age*

- **Age prediction for children.** In our experiments, we noticed that particularly for children the performance of the proposed method was weak, *i.e.* the MAE was significantly larger than for other age groups. Unfortunately, most datasets contain a limited number of children and thus one way of improving would be to collect a dataset of very young people. Alternatively, an expert classifier could be trained dedicated to young people.

- **Improving IMDB-WIKI.** The dataset was crawled automatically from the web. This implies that there are quite a few wrongly labeled images. By manually checking the images, most likely around 3-5% of the images could be re-labeled or removed, which would improve the overall quality of the dataset. Also right now, for each image we took the face with the highest detection score which might not always be the person whose age was inferred from the caption and the date of birth.

- **IMDB-WIKI for aging.** For many celebrities, the dataset contains several hundred images taken over several years. This would allow learning aging patterns specific to a person and thereby allow predictions of how a person will likely look like in the future.

- **Deeper architecture.** Our initial experiments showed that increasing the depth of the neural network increases performance. With the rise of architectures deeper than VGG-16 [134], like Residual Nets [65] with more than 150 layers, these could further improve the performance.

Chapter 5, *Visual guidance for preference prediction*

- **Additional attributes.** So far, the entire model was based on visual features and ratings, predicting the latter. This can be extended by using other attributes (i.e from the Facebook profiles of the users) as features to then also not just predict ratings but also a subset of these attributes. In [148], we explored some of these ideas.

- **Joint optimization.** In the current model the neural network was pre-trained on ImageNet before the matrix factorization algorithm is optimized on top of these features. Thus, both the neural network and the matrix factorization are optimized independently. In future work, one could propose a joint formulation to optimize the weights of the neural network at the same time as the latent factors of the matrix factorization. This could also be of great value to other applications of deep visual features in recommender systems.

- **Reason about attractiveness.** Especially with the rise of `howhot.io`, we often encountered the question to define what attractiveness means based on what the neural network has learned. Unfortunately, nowadays the methods to visualize the weights of a neural network are still limited. Thus beyond the methods of Zeiler and Fergus [179] we did not reveal too many insights of what makes a person attractive. This could be a very interesting piece of work.

Chapter 6, *Deep retrieval for cultural event classification*

- **Consider local information.** Though our approach extracted features at various scales, it is still a holistic approach which considers the entire image. Often the objects in an image tell a lot about the event. By first detecting the objects in an image and then classifying the event based on those or in combination with the current holistic approach could further improve performance.

- **Increase efficiency.** In the current setup of the proposed method, we extract deep features at various scales and encode them into a high-dimensional vector for classification. This results in a relatively slow pipeline. Future work could explore how to speed up the feature extraction and encoding.

8 APPENDIX

The appendix complements the work in Chapter 3. This chapter contains additional experimental results and the derivation of the message passing algorithm for Affinity Propagation Clustering including the 3 extensions proposed: i) the background box; ii) the repellence term; iii) the loss-augmented inference for learning.

8.1 DERIVATION OF MESSAGE PASSING ALGORITHM

In this section we present the derivation of the message passing algorithm for Affinity Propagation Clustering including the proposed extensions.

8.1.1 *Reformulation of global objective function*

We start with the formulation presented in Chapter 3:

$$S_{ij}(c_{ij}) = \begin{cases} s(i,j) & \text{if } c_{ij} = 1 \\ 0 & \text{otherwise,} \end{cases} \tag{8.1}$$

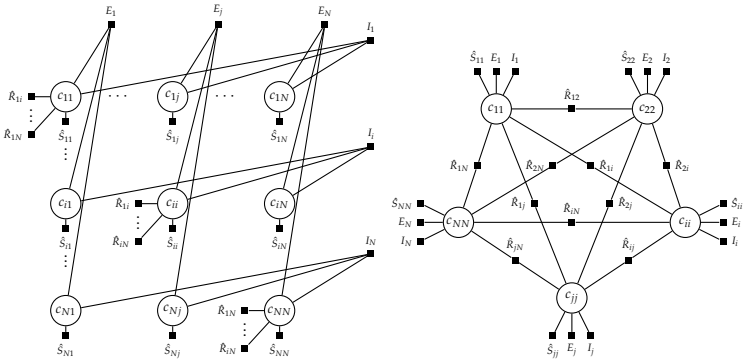
$$\widetilde{I}_i(c_{i1}, ..., c_{iN}) = \begin{cases} -\infty & \text{if } \sum_j c_{ij} > 1 \\ \lambda & \text{if } \sum_j c_{ij} = 0 \text{ (Note that } \lambda = -1) \\ 0 & \text{otherwise,} \end{cases} \tag{8.2}$$

$$E_j(c_{1j}, ..., c_{Nj}) = \begin{cases} -\infty & \text{if } c_{jj} = 0 \text{ and } \exists i \neq j \text{ s.t. } c_{ij} = 1 \\ 0 & \text{otherwise,} \end{cases} \tag{8.3}$$

$$R_{ij}(c_{ii}, c_{jj}) = \begin{cases} r(i,j) & \text{if } c_{ii} = c_{jj} = 1 \\ 0 & \text{otherwise,} \end{cases} \tag{8.4}$$

where we aim to find the labeling which maximizes the following expression

$$\widetilde{E}_{APC} = w_a \sum_i S_{ii} + w_b \sum_{i \neq j} S_{ij} + w_c \sum_i \widetilde{I}_i + w_d \sum_{i<j} R_{ij} + \sum_j E_j. \tag{8.5}$$



(a) Full factor graph (b) Factor graph for diagonal nodes

Figure 8.1: This figure shows the binary variable model for Affinity Propagation with the additional function nodes for repellence ($\hat{R}$). **(a)** shows the full factor graph, whereas **(b)** shows only a subset of the nodes on the diagonal of (a) – these are the only nodes which are connected to $\hat{R}$ terms.

In order to simplify the derivation of the message passing, we incorporate the weights into the local objective functions and model the background box as the the N-th data point[1]. This gives us the following updated similarities

$$\hat{s}(i,j) = \begin{cases} w_a s(i,j) & \text{for } i = j < N \\ -w_c & \text{for } i < N \text{ and } j = N \\ -\infty & \text{for } j < N \text{ and } i = N \\ 0 & \text{for } i = j = N \\ w_b s(i,j) & \text{otherwise,} \end{cases} \tag{8.6}$$

and repellence $\hat{r}(i,j) = w_d r(i,j)$. The max-sum global objective function is then altered to

$$\begin{aligned} \widetilde{E}_{APC}(c_{11},...,c_{ij},...,c_{NN}) = &\sum_{i,j} \hat{S}_{ij}(c_{ij}) + \sum_i I_i(c_{i1},...,c_{iN}) \\ &+ \sum_{i<j} \hat{R}_{ij}(c_{ii},c_{jj}) + \sum_j E_j(c_{1j},...,c_{Nj}) \end{aligned} \tag{8.7}$$

[1] In Chapter 3 the background box was modeled as the N+1-th data point, however to simplify the notation we model it as the N-th data point here.

with the following local objective functions:

$$\hat{S}_{ij}(c_{ij}) = \begin{cases} \hat{s}(i,j) & \text{if } c_{ij} = 1 \\ 0 & \text{otherwise,} \end{cases} \tag{8.8}$$

$$I_i(c_{i1}, ..., c_{iN}) = \begin{cases} -\infty & \text{if } \sum_j c_{ij} \neq 1 \\ 0 & \text{otherwise,} \end{cases} \tag{8.9}$$

$$E_j(c_{1j}, ..., c_{Nj}) = \begin{cases} -\infty & \text{if } c_{jj} = 0 \text{ and } \exists i \neq j \text{ s.t. } c_{ij} = 1 \\ 0 & \text{otherwise,} \end{cases} \tag{8.10}$$

$$\hat{R}_{ij}(c_{ii}, c_{jj}) = \begin{cases} \hat{r}(i,j) & \text{if } c_{ii} = c_{jj} = 1 \\ 0 & \text{otherwise.} \end{cases} \tag{8.11}$$

This is also depicted in Fig. 8.1 as a factor graph.

8.1.2 *Derivation of the messages*

For executing the max-sum algorithm on the factor graph, there are six messages which have to be passed between variable and function nodes. They are shown in Fig. 8.2. As each label c_{ij} is binary we need to send messages with two different values. However, in practice only the difference between the two messages for its two different values needs to be passed. The original messages could still be recovered up to an additive constant, which is not relevant, as this would not affect the optimal assignment. According to [7] the max-sum message update rules are

$$\mu_{x \to f}(x) = \sum_{\{l | f_l \in \text{ne}(x) \setminus f\}} \mu_{f_l \to x}(x), \tag{8.12}$$

$$\mu_{f \to x}(x) = \max_{x_1, ..., x_M} \left(f(x, x_1, \ldots, x_m) + \sum_{\{m | x_m \in \text{ne}(f) \setminus x\}} \mu_{x_m \to f}(x_m) \right). \tag{8.13}$$

Here $\text{ne}(f) \setminus x$ denotes the set of function node f's neighbors excluding variable node x. $\text{ne}(x) \setminus f$ denotes the set of variable nodes x's neighbors excluding function node f. Note that the messages $\alpha_{ij}, \beta_{ij}, \rho_{ij}, \eta_{ij}$ are defined as in the binary formulation of Affinity

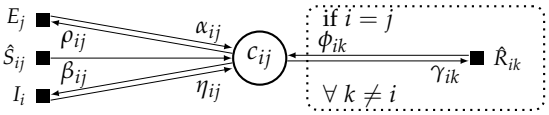


Figure 8.2: The 6 messages passed between variables in our extension of Affinity Propagation are α, β, ρ, η, γ and ϕ.

Propagation presented in [54], *e.g.* $\beta_{ij} = \beta_{ij}(1) - \beta_{ij}(0)$ where $\beta_{ij}(m) = \mu_{c_{ij} \to I_i}(m)$ with $m \in \{0, 1\}$.

We start by deriving the new messages γ and ϕ due to the repellence term. As only the nodes on the diagonal of the factor graph (Fig. 8.1) are connected to the $\hat{R}$ terms, we only have to derive the messages for the case where $i = j$. Let γ_{ik} be the message from c_{ii} to $\hat{R}_{ik}$ and ϕ_{ik} the message from $\hat{R}_{ik}$ to c_{ii}. We evaluate $\phi_{ik}(c_{ii} = \{0, 1\})$ for a given i for all $k \neq i$. This considers the node i and whether it is chosen as an exemplar and its relationship to node k. In the first case i is not chosen as an exemplar. In that case the choice for node k is independent of node i and just the more likely option is chosen:

$$\phi_{ik}(0) = \max(\gamma_{ki}(0), \gamma_{ki}(1)). \tag{8.14}$$

For the case when i is chosen as an exemplar, there is an additional repellence cost $\hat{r}(i, k)$ when choosing node k as an exemplar:

$$\phi_{ik}(1) = \max(\gamma_{ki}(0), \gamma_{ki}(1) + \hat{r}(i, k)). \tag{8.15}$$

Thus combining $\phi_{ik} = \phi_{ik}(1) - \phi_{ik}(0)$ we get

$$\begin{aligned}
\phi_{ik} &= \max(\gamma_{ki}(0), \gamma_{ki}(1) + \hat{r}(i, k)) - \max(\gamma_{ki}(0), \gamma_{ki}(1)) && (8.16)\\
&= (\max(\gamma_{ki}(0), \gamma_{ki}(1) + \hat{r}(i, k)) - \gamma_{ki}(0)) && (8.17)\\
&\quad - (\max(\gamma_{ki}(0), \gamma_{ki}(1)) - \gamma_{ki}(0)) && (8.18)\\
&= \max(0, \gamma_{ki} + \hat{r}(i, k)) - \max(0, \gamma_{ki}). && (8.19)
\end{aligned}$$

Note that in the case when $\hat{r}(i, k) = 0$, we have $\phi_{ik} = 0$ which reduces the extension again to the standard Affinity Propagation formulation. The opposite message γ is as well only defined for the case where $i = j$ for all $k \neq i$. It is derived similarly to the messages β_{ij} and ρ_{ij} in [54] and basically just the sum of all incoming messages except the incoming message by the same function node:

$$\gamma_{ik} = \hat{s}(i, i) + \alpha_{ii} + \eta_{ii} + \sum_{l \notin \{i,k\}} \phi_{il}. \tag{8.20}$$

We skip the derivation of the other 4 original messages $(\beta, \rho, \eta, \alpha)$ at this point as their derivation is nearly equivalent to the derivation presented in [54]. When incorporating the new $\hat{R}$ term we only need to change the messages β and ρ when $i = j$ and their messages are very similar to the original messages. This gives us

$$\beta_{ij} = \begin{cases} \hat{s}(i,j) + \alpha_{ij} & \text{for } i \neq j \\ \hat{s}(i,i) + \alpha_{ii} + \sum_{l \neq i} \phi_{il} & \text{for } i = j \end{cases} \tag{8.21}$$

$$\rho_{ij} = \begin{cases} \hat{s}(i,j) + \eta_{ij} & \text{for } i \neq j \\ \hat{s}(i,i) + \eta_{ii} + \sum_{l \neq i} \phi_{il} & \text{for } i = j. \end{cases} \tag{8.22}$$

The messages α_{ij} and η_{ij} remain unchanged as

$$\eta_{ij} = -\max_{q \neq j}(\beta_{iq}) \tag{8.23}$$

$$\alpha_{ij} = \begin{cases} \sum_{k \neq j} \max(\rho_{kj}, 0) & \text{for } i = j \\ \min(0, \rho_{jj} + \sum_{k \notin \{i,j\}} \max(\rho_{kj}, 0)) & \text{for } i \neq j. \end{cases} \tag{8.24}$$

Now we will reduce the total set of 6 messages to 4 messages by substitution. We will first simplify ρ_{ij} by eliminating η and then β:

$$\rho_{ij} = \begin{cases} \hat{s}(i,j) + \eta_{ij} & \text{for } i \neq j \\ \hat{s}(i,i) + \eta_{ii} + \sum_{l \neq i} \phi_{il} & \text{for } i = j \end{cases} \tag{8.25}$$

$$= \begin{cases} \hat{s}(i,j) - \max_{q \neq j}(\beta_{iq}) & \text{for } i \neq j \\ \hat{s}(i,i) - \max_{q \neq i}(\beta_{iq}) + \sum_{l \neq i} \phi_{il} & \text{for } i = j \end{cases} \tag{8.26}$$

$$= \begin{cases} \hat{s}(i,j) - \max(\max_{q \notin \{i,j\}}(\beta_{iq}), \beta_{ii}) & \text{for } i \neq j \\ \hat{s}(i,i) - \max_{q \neq i}(\beta_{iq}) + \sum_{l \neq i} \phi_{il} & \text{for } i = j \end{cases} \tag{8.27}$$

$$= \begin{cases} \hat{s}(i,j) - \max(\max_{q \notin \{i,j\}}(\hat{s}(i,q) + \alpha_{iq}), \hat{s}(i,i) + \alpha_{ii} + \sum_{l \neq i} \phi_{il}) & \text{for } i \neq j \\ \hat{s}(i,i) - \max_{q \neq i}(\hat{s}(i,q) + \alpha_{iq}) + \sum_{l \neq i} \phi_{il} & \text{for } i = j. \end{cases} \tag{8.28}$$

Similarly for a given note c_{ii} we simplify the messages γ_{ik} for all $k \neq i$ as

$$\gamma_{ik} = \hat{s}(i,i) + \alpha_{ii} + \eta_{ii} + \sum_{l \notin \{i,k\}} \phi_{il} \tag{8.29}$$

$$= \hat{s}(i,i) + \alpha_{ii} - \max_{q \neq i}(\beta_{iq}) + \sum_{l \notin \{i,k\}} \phi_{il} \tag{8.30}$$

$$= \hat{s}(i,i) + \alpha_{ii} - \max_{q \neq i}(\hat{s}(i,q) + \alpha_{iq}) + \sum_{l \notin \{i,k\}} \phi_{il}. \tag{8.31}$$

Thus we eliminated the messages β and η and thereby reduced the total number of messages being passed to 4.

We now have the following 2 messages for all nodes c_{ij}

$$\rho_{ij} = \begin{cases} \hat{s}(i,i) - \max_{q \neq i}(\hat{s}(i,q) + \alpha_{iq}) + \sum_{l \neq i} \phi_{il} & \text{for } i = j \\ \hat{s}(i,j) - \max(\max_{q \notin \{i,j\}}(\hat{s}(i,q) + \alpha_{iq}), \hat{s}(i,i) + \alpha_{ii} + \sum_{l \neq i} \phi_{il}) & \text{for } i \neq j \end{cases} \tag{8.32}$$

$$\alpha_{ij} = \begin{cases} \sum_{k \neq j} \max(\rho_{kj}, 0) & \text{for } i = j \\ \min(0, \rho_{jj} + \sum_{k \notin \{i,j\}} \max(\rho_{kj}, 0)) & \text{for } i \neq j, \end{cases} \tag{8.33}$$

and another 2 messages for all nodes $i = j$ for all $k \neq i$

$$\gamma_{ik} = \hat{s}(i,i) + \alpha_{ii} - \max_{q \neq i}(\hat{s}(i,q) + \alpha_{iq}) + \sum_{l \notin \{i,k\}} \phi_{il} \tag{8.34}$$

$$\phi_{ik} = \max(0, \gamma_{ki} + \hat{r}(i,k)) - \max(0, \gamma_{ki}) \tag{8.35}$$

as shown in Chapter 3.

8.2 MESSAGE PASSING FOR LOSS-AUGMENTED INFERENCE

For the structured-output learning we need to perform loss-augmented inference:

$$\max_{y_n, z_n} \left(\widetilde{E}^n_{APC}(y_n, z_n; \vec{w}) + \Delta(y_n, y^*_n) \right). \tag{8.36}$$

Thus we just need to add another local objective function Δ to the global objective function $\widetilde{E}^n_{APC}$ which is defined as

$$\Delta_{ij}(c_{ij}) = \begin{cases} \nu & \text{for } i=j \text{ and } c_{ii}=0 \text{ and } c^n_{ii}=1 \\ \pi\left(1 - \max_{\text{obj}} \frac{|i \cap obj|}{|i \cup obj|}\right) & \text{for } i=j \text{ and } c_{ii}=1 \text{ and } c^n_{ii}=0 \\ 0 & \text{otherwise,} \end{cases} \tag{8.37}$$

where c^n_{ii} is the ground truth labeling for image n as described in Chapter 3. We will now combine the two local objective function Δ and $\hat{S}$ into a combined local function $\hat{S}_\Delta$ which results in

$$\hat{S}_{\Delta ij}(c_{ij}) = \begin{cases} \nu & \text{for } i=j \text{ and } c_{ii}=0 \text{ and } c^n_{ii}=1 \\ \hat{s}(i,j) + \pi\left(1 - \max_{\text{obj}} \frac{|i \cap obj|}{|i \cup obj|}\right) & \text{for } i=j \text{ and } c_{ii}=1 \text{ and } c^n_{ii}=0 \\ \hat{s}(i,j) & \text{for } c_{ij}=1 \text{ otherwise} \\ 0 & \text{otherwise.} \end{cases} \tag{8.38}$$

We rederive the message coming from $\hat{S}_{\Delta ij}$ denoted as $\hat{s}_{\Delta ij}$. Setting c_{ij} to 0 and 1 respectively yields

$$\hat{s}_{\Delta ij}(0) = \begin{cases} \nu & \text{for } i=j \text{ and } c^n_{ii}=1 \\ 0 & \text{otherwise,} \end{cases} \tag{8.39}$$

$$\hat{s}_{\Delta ij}(1) = \begin{cases} \hat{s}(i,j) + \pi\left(1 - \max_{\text{obj}} \frac{|i \cap obj|}{|i \cup obj|}\right) & \text{for } i=j \text{ and } c^n_{ii}=0 \\ \hat{s}(i,j) & \text{otherwise.} \end{cases} \tag{8.40}$$

Now taking the difference $\hat{s}_\Delta(i,j) = \hat{s}_{\Delta ij}(1) - \hat{s}_{\Delta ij}(0)$, we arrive at the final updated message from the combined node of the two local functions as

$$\hat{s}_\Delta(i,j) = \begin{cases} \hat{s}(i,j) - \nu & \text{for } i=j \text{ and } c^n_{ii}=1 \\ \hat{s}(i,j) + \pi\left(1 - \max_{\text{obj}} \frac{|i \cap obj|}{|i \cup obj|}\right) & \text{for } i=j \text{ and } c^n_{ii}=0 \\ \hat{s}(i,j) & \text{otherwise.} \end{cases} \tag{8.41}$$

Note that the outgoing message is of no interest as $\hat{s}_\Delta$ does not depend on it. Thus in order to perform loss augmented inference, we just need to alter the similarities from $\hat{s}$ to $\hat{s}_\Delta$.

8.3 OBJECT CLASS DETECTION RESULTS

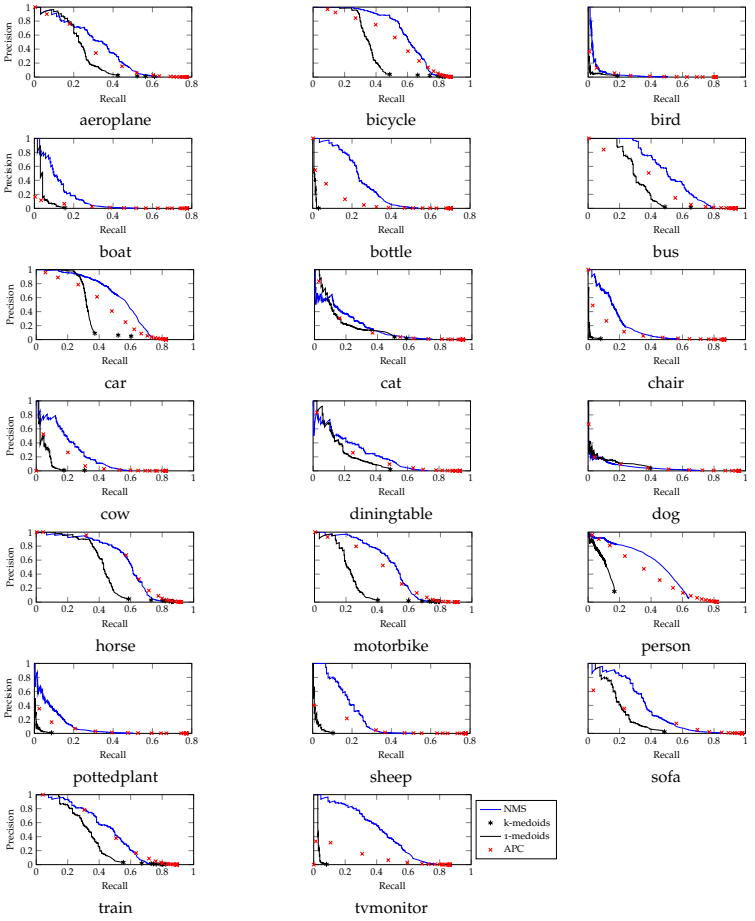


Figure 8.3: Precision vs. recall plots for *IoU* 0.5 for all classes

BIBLIOGRAPHY

[1] M. Aharon, M. Elad, and A. Bruckstein, "K-svd: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Transactions on signal processing*, vol. 54, no. 11, pp. 4311–4322, 2006 (cit. on p. 12).

[2] B. Alexe, T. Deselaers, and V. Ferrari, "Measuring the objectness of image windows," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 11, pp. 2189–2202, 2012 (cit. on pp. 19, 22, 38).

[3] B. Alexe, T. Deselaers, and V. Ferrari, "What is an object?" In *IEEE Conference on Computer Vision and Pattern Recognition*, 2010 (cit. on p. 37).

[4] H. Altwaijry and S. Belongie, "Relative ranking of facial attractiveness," in *IEEE Winter Conference on Applications of Computer Vision*, 2013 (cit. on p. 76).

[5] O. Barinova, V. Lempitsky, and P. Kholi, "On detection of multiple object instances using hough transforms," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 9, pp. 1773–1784, 2012 (cit. on p. 22).

[6] X. Baro, J. Gonzalez, J. Fabian, M. A. Bautista, M. Oliu, H. Jair Escalante, I. Guyon, and S. Escalera, "ChaLearn Looking at People 2015 Challenges: Action Spotting and Cultural Event Recognition," in *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2015 (cit. on pp. 99, 112, 113).

[7] C. M. Bishop, *Pattern recognition and machine learning*. Elsevier, 2006 (cit. on p. 123).

[8] M. B. Blaschko, "Branch and Bound Strategies for Non-maximal Suppression in Object Detection," in *International Workshop on Energy Minimization Methods in Computer Vision and Pattern Recognition*, 2013 (cit. on p. 22).

[9] M. B. Blaschko, J. Kannala, and E. Rahtu, "Non Maximal Suppression in Cascaded Ranking Models," in *Scandinavian Conference on Image Analysis*, 2013 (cit. on p. 22).

[10] M. B. Blaschko and C. H. Lampert, "Learning to localize objects with structured output regression," in *IEEE European Conference on Computer Vision*, 2008 (cit. on p. 22).

[11] J. S. Breese, D. Heckerman, and C. Kadie, "Empirical analysis of predictive algorithms for collaborative filtering," in *Conference on Uncertainty in Artificial Intelligence*, 1998 (cit. on p. 76).

[12] L. Brozovsky and V. Petricek, "Recommender system for online dating service," *arXiv preprint cs/0703042*, 2007 (cit. on p. 77).

[13] D. Cai, X. He, and J. Han, "SRDA: An efficient algorithm for large-scale discriminant analysis," *Transactions on Knowledge and Data Engineering*, vol. 20, no. 1, pp. 1–12, 2008 (cit. on p. 103).

[14] J. Canny, "A computational approach to edge detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 8, no. 6, pp. 679–698, 1986 (cit. on pp. 19, 21, 39).

[15] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, pp. 1–27, 3 2011 (cit. on p. 78).

[16] K.-Y. Chang, C.-S. Chen, and Y.-P. Hung, "Ordinal hyperplanes ranker with cost sensitivities for age estimation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2011 (cit. on pp. 45, 56, 64, 75, 78, 79).

[17] B.-C. Chen, C.-S. Chen, and W. H. Hsu, "Face Recognition and Retrieval Using Cross-Age Reference Coding With Cross-Age Celebrity Dataset," *IEEE Transactions on Multimedia*, vol. 17, no. 6, pp. 804–815, 2015 (cit. on pp. 41, 44, 57–59, 64, 65).

[18] G. Chen, Y. Ding, J. Xiao, and T. X. Han, "Detection evolution with multi-order contextual co-occurrence," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2013 (cit. on p. 22).

[19] J.-C. Chen, V. M. Patel, and R. Chellappa, "Unconstrained Face Verification using Deep CNN Features," in *IEEE Winter Conference on Applications of Computer Vision*, 2016 (cit. on p. 47).

[20] K. Chen, S. Gong, T. Xiang, and C. Change Loy, "Cumulative Attribute Space for Age and Crowd Density Estimation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2013 (cit. on pp. 45, 56, 64, 75, 78, 79).

[21] M.-M. Cheng, Z. Zhang, W.-Y. Lin, and P. H. S. Torr, "BING: Binarized Normed Gradients for Objectness Estimation at 300fps," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2014 (cit. on p. 19).

[22] R. G. Cinbis, J. Verbeek, and C. Schmid, "Segmentation Driven Object Detection with Fisher Vectors," in *IEEE International Conference on Computer Vision*, 2013 (cit. on p. 22).

[23] D. Ciregan, U. Meier, and J. Schmidhuber, "Multi-column Deep Neural Networks for Image Classification," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2012 (cit. on pp. 42, 97).

[24] T. F. Cootes, G. J. Edwards, and C. J. Taylor, "Active Appearance Models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 6, pp. 681–685, 2001 (cit. on p. 45).

[25] C. Cortes and V. Vapnik, "Support-vector networks," *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995 (cit. on p. 45).

[26] D. Dai, Y. Wang, Y. Chen, and L. Van Gool, "Is Image Super-resolution Helpful for Other Vision Tasks?" In *IEEE Winter Conference on Applications of Computer Vision*, 2016 (cit. on pp. 9, 18).

[27] N. Dalal, "Finding people in images and videos," PhD thesis, Institut National Polytechnique de Grenoble-INPG, 2006 (cit. on p. 22).

[28] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2005 (cit. on pp. 19, 21).

[29] C. Desai, D. Ramanan, and C. C. Fowlkes, "Discriminative models for multi-class object layout," *International Journal of Computer Vision*, vol. 95, no. 1, pp. 1–12, 2011 (cit. on p. 23).

[30] S. Dhar, V. Ordonez, and T. L. Berg, "High level describable attributes for predicting aesthetics and interestingness," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2011 (cit. on p. 76).

[31] Y. Ding and J. Xiao, "Contextual boost for pedestrian detection," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2012 (cit. on p. 22).

[32] P. Doll and C. L. Zitnick, "Structured Forests for Fast Edge Detection," in *IEEE International Conference on Computer Vision*, 2013 (cit. on p. 39).

[33] H. Drucker, C. J. C. Burges, L. Kaufman, A. J. Smola, and V. Vapnik, "Support Vector Regression Machines," in *Advances in Neural Information Processing Systems*, 1997 (cit. on p. 45).

[34] D. Dueck and B. J. Frey, "Non-metric affinity propagation for unsupervised image categorization," in *IEEE International Conference on Computer Vision*, 2007 (cit. on p. 23).

[35] D. Dueck, B. J. Frey, N. Jojic, V. Jojic, G. Giaever, A. Emili, G. Musso, and R. Hegele, "Using Affinity Propagation," in *International Conference on Research in Computational Molecular Biology*, 2008 (cit. on p. 23).

[36] E. Eidinger, R. Enbar, and T. Hassner, "Age and Gender Estimation of Unfiltered Faces," *IEEE Transactions on Information Forensics and Security*, vol. 9, no. 12, pp. 2170–2179, 2014 (cit. on pp. 41, 44, 47, 65).

[37] Y. Eisenthal, G. Dror, and E. Ruppin, "Facial attractiveness: Beauty and the machine," *Neural Computation*, vol. 18, no. 1, pp. 119–142, 2006 (cit. on p. 76).

[38] S. Escalera, J. Fabian, P. Pardo, X. Baro, J. Gonzalez, H. J. Escalante, D. Misevic, U. Steiner, and I. Guyon, "ChaLearn Looking at People 2015: Apparent Age and Cultural Event Recognition Datasets and Results," in *IEEE International Conference on Computer Vision Workshops*, 2015 (cit. on pp. 3, 4, 8, 41–43, 46, 57–61, 63, 97, 99, 105, 116).

[39] 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*, vol. 88, no. 2, pp. 303–338, 2010 (cit. on pp. 19, 31, 97).

[40] L. G. Farkas and S. A. Schendel, "Anthropometry of the Head and Face," *American Journal of Orthodontics and Dentofacial Orthopedics*, vol. 107, no. 1, pp. 112–112, 1995 (cit. on p. 45).

[41] S. L. Feld, "Why Your Friends Have More Friends Than You Do," *American Journal of Sociology*, vol. 96, no. 6, pp. 1464–1477, 1991 (cit. on p. 87).

[42] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object Detection with Discriminatively Trained Part-Based Models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 9, pp. 1627–1645, 2010 (cit. on pp. 19, 21, 23, 31, 48, 117).

[43] A. Foi, V. Katkovnik, and K. Egiazarian, "Pointwise Shape-Adaptive DCT for High-Quality Denoising and Deblocking of Grayscale and Color Images," *IEEE Transactions on Image Processing*, vol. 16, no. 5, pp. 1395–1411, 2007 (cit. on p. 10).

[44] B. J. Frey and D. Dueck, "Clustering by passing messages between data points," *Science*, vol. 315, 2007 (cit. on pp. 3, 6, 21, 23, 24, 28, 115).

[45] J. H. Friedman, "Regularized discriminant analysis," *Journal of the American statistical association*, vol. 84, no. 405, pp. 165–175, 1989 (cit. on pp. 98, 103).

[46] Y. Fu, G. Guo, and T. S. Huang, "Age Synthesis and Estimation via Faces: A Survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 11, pp. 1955–1976, 2010 (cit. on pp. 44, 75).

[47] Y. Fu and T. S. Huang, "Human Age Estimation With Regression on Discriminative Aging Manifold," *IEEE Transactions on Multimedia*, vol. 10, no. 4, pp. 578–584, 2008 (cit. on p. 45).

[48] H. Fukai, H. Takimoto, Y. Mitsukura, and M. Fukumi, "Apparent age estimation system based on age perception," in *SICE Annual Conference*, 2007 (cit. on p. 45).

[49] F. Gao and H. Ai, "Face age classification on consumer images with gabor feature and fuzzy lda method," in *International Conference on Biometrics*, 2009 (cit. on p. 45).

[50] P. Geladi and B. R. Kowalski, "Partial least-squares regression: a tutorial," *Analytica Chimica Acta*, vol. 185, pp. 1–17, 1986 (cit. on p. 45).

[51] X. Geng, Z.-H. Zhou, and K. Smith-Miles, "Automatic Age Estimation Based on Facial Aging Patterns," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 12, pp. 2234–2240, 2007 (cit. on pp. 45, 64, 79).

[52] R. B. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2014 (cit. on pp. 19, 42, 75, 78, 97, 102, 117).

[53] I. E. Givoni, C. Chung, and B. J. Frey, "Hierarchical Affinity Propagation," *arXiv preprint arXiv:1202.3722*, 2012 (cit. on p. 23).

[54] I. E. Givoni and B. J. Frey, "A Binary Variable Model for Affinity Propagation," *Neural Computation*, vol. 21, no. 6, pp. 1589–1600, 2009 (cit. on pp. 23, 24, 28, 124, 125).

[55] I. E. Givoni and B. J. Frey, "Semi-Supervised Affinity Propagation with Instance-Level Constraints," in *International Conference on Artificial Intelligence and Statistics*, 2009 (cit. on p. 23).

[56] D. Gray, K. Yu, W. Xu, and Y. Gong, "Predicting facial beauty without landmarks," in *IEEE European Conference on Computer Vision*, 2010 (cit. on pp. 76, 78–80).

[57] G. Guo, "Video Analytics for Business Intelligence," in. Springer, 2012, ch. Human Age Estimation and Sex Classification, pp. 101–131 (cit. on pp. 41, 44, 45).

[58] G. Guo, Y. Fu, C. R. Dyer, and T. S. Huang, "Image-Based Human Age Estimation by Manifold Learning and Locally Adjusted Robust Regression," *IEEE Transactions on Image Processing*, vol. 17, no. 7, pp. 1178–1188, 2008 (cit. on pp. 45, 56, 64, 78, 79).

[59] G. Guo and G. Mu, "A framework for joint estimation of age, gender and ethnicity on a large database," *Image and Vision Computing*, vol. 32, no. 10, pp. 761–770, 2014 (cit. on pp. 45, 57, 64).

[60] G. Guo and G. Mu, "Joint estimation of age, gender and ethnicity: CCA vs. PLS," in *IEEE International Conference on Automatic Face and Gesture Recognition*, 2013 (cit. on p. 78).

[61] G. Guo and G. Mu, "Simultaneous dimensionality reduction and human age estimation via kernel partial least squares regression," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2011 (cit. on pp. 45, 64, 78).

[62] H. Han, C. Otto, and A. K. Jain, "Age estimation from face images: Human vs. machine performance," in *International Conference on Biometrics*, 2013 (cit. on pp. 41, 75).

[63] H. Han, C. Otto, X. Liu, and A. K. Jain, "Demographic Estimation from Face Images: Human vs. Machine Performance," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 6, pp. 1148–1161, 2015 (cit. on pp. 44, 45, 64).

[64] D. R. Hardoon, S. Szedmak, and J. Shawe-Taylor, "Canonical Correlation Analysis: An Overview with Application to Learning Methods," *Neural Computation*, vol. 16, no. 12, pp. 2639–2664, 2004 (cit. on p. 45).

[65] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016 (cit. on pp. 42, 70, 118).

[66] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," in *IEEE European Conference on Computer Vision*, 2014 (cit. on p. 102).

[67] M. Hoai, "Regularized max pooling for image categorization," in *British Machine Vision Conference*, 2014 (cit. on p. 100).

[68] D. Hoiem, Y. Chodpathumwan, and Q. Dai, "Diagnosing Error in Object Detectors," in *IEEE European Conference on Computer Vision*, 2012 (cit. on p. 30).

[69] G. B. Huang, V. Jain, and E. Learned-Miller, "Unsupervised joint alignment of complex images," in *IEEE International Conference on Computer Vision*, 2007 (cit. on p. 80).

[70] I. Huerta, C. Fernández, and A. Prati, "Facial Age Estimation Through the Fusion of Texture and Local Appearance Descriptors," in *IEEE European Conference on Computer Vision*, 2014 (cit. on pp. 45, 57, 64).

[71] A. Jaimes and N. Sebe, "Multimodal human–computer interaction: A survey," *Computer Vision and Image Understanding*, vol. 108, no. 1, pp. 116–134, 2007 (cit. on p. 75).

[72] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional Architecture for Fast Feature Embedding," in *International Conference on Multimedia*, 2014 (cit. on pp. 54, 106).

[73] L. Kaufman and P. Rousseeuw, "Clustering by means of medoids," *Statistical Data Analysis Based on the L1-Norm and Related Methods*, 1987 (cit. on p. 23).

[74] V. Kazemi and J. Sullivan, "One Millisecond Face Alignment with an Ensemble of Regression Trees," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2014 (cit. on p. 47).

[75] M. H. Kiapour, K. Yamaguchi, A. C. Berg, and T. L. Berg, "Hipster wars: Discovering elements of fashion styles," in *IEEE European Conference on Computer Vision*, 2014 (cit. on p. 76).

[76] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *Computer*, vol. 42, no. 8, pp. 30–37, 2009 (cit. on pp. 74, 76, 81).

[77] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," in *Advances in Neural Information Processing Systems*, 2012 (cit. on pp. 1, 42, 49, 75, 97, 99).

[78] A. Krzywicki, W. Wobcke, X. Cai, A. Mahidadia, M. Bain, P. Compton, and Y. S. Kim, "Interaction-based collaborative filtering methods for recommendation in online dating," in *International Conference on Web Information Systems Engineering*, 2010 (cit. on p. 77).

[79] N. Kumar, A. C. Berg, P. N. Belhumeur, and S. K. Nayar, "Attribute and simile classifiers for face verification," in *IEEE International Conference on Computer Vision*, 2009 (cit. on p. 75).

[80] H. Kwon, K. Yun, M. Hoai, and D. Samaras, "Recognizing Cultural Events in Images: A Study of Image Categorization Models," in *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2015 (cit. on pp. 100, 113).

[81] Y. H. Kwon and N. da Vitoria Lobo, "Age Classification from Facial Images," *Computer Vision and Image Understanding*, vol. 74, no. 1, pp. 1–21, 1999 (cit. on p. 45).

[82] Y. Kwon, K. I. Kim, J. Tompkin, J. H. Kim, and C. Theobalt, "Efficient Learning of Image Super-resolution and Compression Artifact Removal with Semi-local Gaussian Processes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 9, pp. 1792–1805, 2015 (cit. on pp. 3, 5, 10, 11, 13, 14, 16–18, 115).

[83] L. Ladickỳ, P. Sturgess, K. Alahari, C. Russell, and P. H. Torr, "What, where and how many? combining object detectors and crfs," in *IEEE European Conference on Computer Vision*, 2010 (cit. on p. 23).

[84] A. Lanitis, C. Draganova, and C. Christodoulou, "Comparing different classifiers for automatic age estimation," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 34, no. 1, pp. 621–628, 2004 (cit. on p. 45).

[85] V. Laparra, J. Gutiérrez, G. Camps-Valls, and J. Malo, "Image Denoising with Kernels Based on Natural Image Relations," *Journal of Machine Learning Research*, vol. 11, no. Feb, pp. 873–903, 2010 (cit. on p. 10).

[86] A. Laurentini and A. Bottino, "Computer analysis of face beauty: A survey," *Computer Vision and Image Understanding*, vol. 125, pp. 184–199, 2014 (cit. on p. 76).

[87] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2006 (cit. on pp. 100, 102).

[88] N. Lazic, B. J. Frey, and P. Aarabi, "Solving the Uncapacitated Facility Location Problem Using Message Passing Algorithms," in *International Conference on Artificial Intelligence and Statistics*, 2010 (cit. on p. 23).

[89] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015 (cit. on p. 1).

[90] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998 (cit. on p. 45).

[91] D. D. Lee and H. S. Seung, "Algorithms for non-negative matrix factorization," in *Advances in Neural Information Processing Systems*, 2001 (cit. on p. 76).

[92] G. Levi and T. Hassner, "Age and gender classification using convolutional neural networks," in *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2015 (cit. on pp. 45, 65).

[93] T. Leyvand, D. Cohen-Or, G. Dror, and D. Lischinski, "Data-driven enhancement of facial attractiveness," *ACM Transactions on Graphics*, vol. 27, no. 3, 2008 (cit. on p. 90).

[94] X. Li, "Improved wavelet decoding via set theoretic estimation," *IEEE transactions on circuits and systems for video technology*, vol. 15, no. 1, pp. 108–112, 2005 (cit. on p. 10).

[95] L. Liu, J. Xing, S. Liu, H. Xu, X. Zhou, and S. Yan, "Wow! You Are So Beautiful Today!" *ACM Transactions on Multimedia Computing, Communications, and Applications*, vol. 11, no. 1s, pp. 1–22, 2014 (cit. on p. 90).

[96] X. Liu, S. Li, M. Kan, J. Zhang, S. Wu, W. Liu, H. Han, S. Shan, and X. Chen, "AgeNet: Deeply Learned Regressor and Classifier for Robust Apparent Age Estimation," in *IEEE International Conference on Computer Vision Workshops*, 2015 (cit. on pp. 46, 63).

[97] K. Luu, K. Ricanek, T. D. Bui, and C. Y. Suen, "Age estimation using Active Appearance Models and Support Vector Machine regression," in *IEEE International Conference on Biometrics: Theory, Applications, and Systems*, 2009 (cit. on pp. 45, 64).

[98] K. Luu, K. Seshadri, M. Savvides, T. D. Bui, and C. Y. Suen, "Contourlet appearance model for facial age estimation," in *International Joint Conference on Biometrics*, 2011 (cit. on p. 64).

[99] L. van der Maaten and G. E. Hinton, "Visualizing High-Dimensional Data Using t-SNE," *Journal of Machine Learning Research*, vol. 9, no. Nov, pp. 2579–2605, 2008 (cit. on p. 68).

[100] J. MacQueen *et al.*, "Some methods for classification and analysis of multivariate observations," in *Berkeley Symposium on Mathematical Statistics and Probability*, 1967 (cit. on p. 23).

[101] S. Manén, M. Guillaumin, and L. Van Gool, "Prime Object Proposals with Randomized Prim's Algorithm," in *IEEE International Conference on Computer Vision*, 2013 (cit. on p. 37).

[102] L. Marchesotti, N. Murray, and F. Perronnin, "Discovering Beautiful Attributes for Aesthetic Image Analysis," *International Journal of Computer Vision*, vol. 113, no. 3, pp. 246–266, 2015 (cit. on p. 76).

[103] M. Mathias, R. Benenson, M. Pedersoli, and L. Van Gool, "Face Detection without Bells and Whistles," in *IEEE European Conference on Computer Vision*, 2014 (cit. on pp. 43, 47, 87).

[104] K. Mikolajczyk and C. Schmid, "Scale & Affine Invariant Interest Point Detectors," *International Journal of Computer Vision*, vol. 60, no. 1, pp. 63–86, 2004 (cit. on p. 21).

[105] G. Mu, G. Guo, Y. Fu, and T. S. Huang, "Human age estimation using bio-inspired features," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2009 (cit. on pp. 45, 90).

[106] Y. Mu, "Computational facial attractiveness prediction by aesthetics-aware features," *Neurocomputing*, vol. 99, pp. 59–64, 2013 (cit. on p. 76).

[107] A. V. Nasonov and A. S. Krylov, "Scale-space method of image ringing estimation.," in *IEEE International Conference on Image Processing*, 2009 (cit. on p. 9).

[108] C. B. Ng, Y. H. Tay, and B.-M. Goi, "Recognizing human gender in computer vision: a survey," in *Pacific Rim International Conference on Artificial Intelligence*, 2012 (cit. on p. 75).

[109] A. Nosratinia, "Enhancement of JPEG-Compressed Images by Re-application of JPEG," *Journal of VLSI signal processing systems for signal, image and video technology*, vol. 27, no. 1-2, pp. 69–79, 2001 (cit. on p. 10).

[110] A. Nosratinia, "Postprocessing of JPEG-2000 images to remove compression artifacts," *IEEE Signal Processing Letters*, vol. 10, no. 10, pp. 296–299, 2003 (cit. on pp. 10, 14).

[111] G. Panis, A. Lanitis, N. Tsapatsoulis, and T. F. Cootes, "Overview of research on facial ageing using the FG-NET ageing database," *IET Biometrics*, vol. 5, no. 2, pp. 37–46, 2016 (cit. on pp. 41, 42, 44, 56, 58, 59, 64).

[112] S. Park and N. Kwak, "Cultural Event Recognition by Subregion Classification With Convolutional Neural Network," in *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2015 (cit. on pp. 99, 113).

[113] G. Qiu, "MLP for adaptive postprocessing block-coded images," *IEEE transactions on circuits and systems for video technology*, vol. 10, no. 8, pp. 1450–1454, 2000 (cit. on p. 10).

[114] N. Ramanathan and R. Chellappa, "Modeling Age Progression in Young Faces," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2006 (cit. on p. 45).

[115] N. Razavi, J. Gall, and L. Van Gool, "Backprojection Revisited: Scalable Multi-view Object Detection and Similarity Metrics for Detections," in *IEEE European Conference on Computer Vision*, 2010 (cit. on p. 22).

[116] A. S. Razavian, H. Azizpour, J. Sullivan, and S. Carlsson, "CNN Features off-the-shelf: an Astounding Baseline for Recognition," in *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2014 (cit. on p. 77).

[117] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Advances in Neural Information Processing Systems*, 2015 (cit. on pp. 1, 117).

[118] K. Ricanek and T. Tesafaye, "MORPH: a longitudinal image database of normal adult age-progression," in *Automatic Face and Gesture Recognition*, 2006 (cit. on pp. 41, 44, 56, 58, 59, 64, 78, 79).

[119] M. Ristin, J. Gall, and L. Van Gool, "Local context priors for object proposal generation," in *Asian Conference on Computer Vision*, 2012 (cit. on p. 37).

[120] G. B. Robinson, *Automated collaborative filtering in world wide web advertising*, US Patent 5,918,014, 1999 (cit. on p. 74).

[121] S. Roth and M. J. Black, "Fields of experts," *International Journal of Computer Vision*, vol. 82, no. 2, pp. 205–229, 2009 (cit. on pp. 10, 14, 16).

[122] R. Rothe, M. Guillaumin, and L. Van Gool, "Non-Maximum Suppression for Object Detection by Passing Messages between Windows," in *Asian Conference on Computer Vision*, 2014 (cit. on pp. 4, 6).

[123] R. Rothe, M. Ristin, M. Dantone, and L. Van Gool, "Discriminative Learning of Apparel Features," in *Conference on Machine Vision Applications*, 2015 (cit. on p. 5).

[124] R. Rothe, R. Timofte, and L. Van Gool, "Deep expectation of real and apparent age from a single image without facial landmarks," *International Journal of Computer Vision*, 2016 (cit. on pp. 4, 7, 46).

[125] R. Rothe, R. Timofte, and L. Van Gool, "DEX: Deep EXpectation of Apparent Age From a Single Image," in *IEEE International Conference on Computer Vision Workshops*, 2015 (cit. on pp. 5, 44, 46, 63, 75, 91).

[126] R. Rothe, R. Timofte, and L. Van Gool, "DLDR: Deep Linear Discriminative Retrieval for cultural event classification from a single image," in *IEEE International Conference on Computer Vision Workshops*, 2015 (cit. on pp. 4, 8).

[127] R. Rothe, R. Timofte, and L. Van Gool, "Efficient regression priors for reducing image compression artifacts," in *IEEE International Conference on Image Processing*, 2015 (cit. on pp. 4, 6).

[128] R. Rothe, R. Timofte, and L. Van Gool, "Some like it hot - visual guidance for preference prediction," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016 (cit. on pp. 4, 7, 42, 45, 56, 63, 64).

[129] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet Large Scale Visual Recognition Challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, 2015 (cit. on pp. 42, 43, 49, 97, 100).

[130] A. Salvador, M. Zeppelzauer, D. Manchon-Vizuete, A. Calafell, and X. Giro-i Nieto, "Cultural Event Recognition with Visual ConvNets and Temporal Models," in *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2015 (cit. on pp. 99, 113).

[131] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, "Item-based collaborative filtering recommendation algorithms," in *International Conference on World Wide Web*, 2001 (cit. on p. 76).

[132] H. Schneiderman and T. Kanade, "Object detection using the statistics of parts," *International Journal of Computer Vision*, vol. 56, no. 3, pp. 151–177, 2004 (cit. on p. 21).

[133] Y. Shi, M. Larson, and A. Hanjalic, "Collaborative filtering beyond the user-item matrix: A survey of the state of the art and future challenges," *ACM Computing Surveys*, vol. 47, no. 1, p. 3, 2014 (cit. on pp. 76, 77, 81).

[134] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," in *International Conference on Learning Representations*, 2015 (cit. on pp. 7, 8, 43, 49, 54, 75, 78, 97, 99, 100, 118).

[135] K. Sugiyama, K. Hatano, and M. Yoshikawa, "Adaptive Web Search Based on User Profile Constructed Without Any Effort from Users," in *International Conference on World Wide Web*, 2004 (cit. on p. 77).

[136] J. Suo, S.-C. Zhu, S. Shan, and X. Chen, "A Compositional and Dynamic Model for Face Aging," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 3, pp. 385–401, 2010 (cit. on p. 45).

[137] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015 (cit. on pp. 46, 47, 99).

[138] C. Szegedy, A. Toshev, and D. Erhan, "Deep Neural Networks for Object Detection," in *Advances in Neural Information Processing Systems*, 2013 (cit. on pp. 22, 37).

[139] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "DeepFace: Closing the Gap to Human-Level Performance in Face Verification," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2014 (cit. on p. 48).

[140] S. Tang, M. Andriluka, and B. Schiele, "Detection and tracking of occluded people," *International Journal of Computer Vision*, vol. 110, no. 1, pp. 58–69, 2014 (cit. on p. 23).

[141] R. Timofte, V. De Smet, and L. Van Gool, "A+: Adjusted Anchored Neighborhood Regression for Fast Super-Resolution," in *Asian Conference on Computer Vision*, 2014 (cit. on pp. 3, 5, 10–12, 16, 83, 115).

[142] R. Timofte, V. De Smet, and L. Van Gool, "Anchored Neighborhood Regression for Fast Example-Based Super-Resolution," in *IEEE International Conference on Computer Vision*, 2013 (cit. on pp. 11, 12).

[143] R. Timofte, V. De Smet, and L. Van Gool, "Semantic super-resolution: When and where is it useful?" *Computer Vision and Image Understanding*, vol. 142, pp. 1–12, 2016 (cit. on p. 117).

[144] R. Timofte, R. Rothe, and L. Van Gool, "Seven ways to improve example-based single image super resolution," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016 (cit. on p. 5).

[145] R. Timofte and L. Van Gool, "Iterative Nearest Neighbors," *Pattern Recognition*, vol. 48, no. 1, pp. 60–72, 2015 (cit. on pp. 98, 103).

[146] R. Timofte and L. Van Gool, "Iterative nearest neighbors for classification and dimensionality reduction," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2012 (cit. on pp. 4, 8, 98, 103, 116).

[147] R. Timofte and L. Van Gool, "Sparse representation based projections," in *British Machine Vision Conference*, 2011 (cit. on p. 103).

[148] R. Torfason, E. Agustsson, R. Rothe, and R. Timofte, "From face images and attributes to attributes," in *Asian Conference on Computer Vision*, 2016 (cit. on pp. 5, 119).

[149] D. Tschumperle and R. Deriche, "Vector-valued image regularization with PDEs: a common framework for different applications," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 4, pp. 506–517, 2005 (cit. on p. 10).

[150] J. R. Uijlings, K. E. van de Sande, T. Gevers, and A. W. Smeulders, "Selective Search for Object Recognition," *International Journal of Computer Vision*, vol. 104, no. 2, pp. 154–171, 2013 (cit. on pp. 22, 37).

[151] M. Uricar, R. Timofte, R. Rothe, J. Matas, and L. Van Gool, "Structured Output SVM Prediction of Apparent Age, Gender and Smile From Deep Features," in *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2016 (cit. on pp. 5, 45).

[152] V. N. Vapnik, *Statistical Learning Theory*. Wiley New York, 1998 (cit. on p. 75).

[153] A. Vedaldi, *A MATLAB wrapper of* SVM$^{\text{struct}}$, 2011 (cit. on p. 29).

[154] A. Vedaldi and K. Lenc, "MatConvNet – Convolutional Neural Networks for MATLAB," in *International Conference on Multimedia*, 2015 (cit. on p. 78).

[155] P. Viola and M. Jones, "Robust real-time object detection," *International Journal of Computer Vision*, vol. 4, 2001 (cit. on pp. 19, 21).

[156] U. Von Luxburg, "A tutorial on spectral clustering," *Statistics and computing*, vol. 17, no. 4, pp. 395–416, 2007 (cit. on p. 23).

[157] L. Wang, S. Guo, W. Huang, and Y. Qiao, "Places205-VGGNet Models for Scene Recognition," *arXiv preprint arXiv:1508.01667*, 2015 (cit. on p. 101).

[158] L. Wang, Z. Wang, W. Du, and Y. Qiao, "Object-Scene Convolutional Neural Networks for Event Recognition in Images," in *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2015 (cit. on pp. 99, 101, 113).

[159] X. Wang, R. Guo, and C. Kambhamettu, "Deeply-Learned Feature for Age Estimation," in *IEEE Winter Conference on Applications of Computer Vision*, 2015 (cit. on pp. 45, 56, 63, 64, 75, 78, 79).

[160] J. Whitehill and J. R. Movellan, "Personalized facial attractiveness prediction," in *IEEE International Conference on Automatic Face and Gesture Recognition*, 2008 (cit. on p. 76).

[161] J. Willis and A. Todorov, "First impressions: Making up your mind after 100 ms exposure to a face," *Psychological science*, vol. 17, no. 7, pp. 592–598, 2006 (cit. on p. 73).

[162] P. Wohlhart, M. Donoser, P. M. Roth, and H. Bischof, "Detecting partially occluded objects with an implicit shape model random field," in *Asian Conference on Computer Vision*, 2012 (cit. on p. 22).

[163] W. Wojcikiewicz, "Probabilistic modelling of multiple observations in face detection," Humboldt-Universität zu Berlin, Tech. Rep., 2008 (cit. on p. 22).

[164] B. Wu and R. Nevatia, "Detection and segmentation of multiple, partially occluded objects by grouping, merging, assigning part detection responses," *International Journal of Computer Vision*, vol. 82, no. 2, pp. 185–204, 2009 (cit. on p. 22).

[165] J. Wu, R. Timofte, and L. Van Gool, "Efficient Regression Priors for post-processing demosaiced images," in *IEEE International Conference on Image Processing*, 2015 (cit. on p. 10).

[166] Z. Xu, H. Chen, S.-C. Zhu, and J. Luo, "A Hierarchical Compositional Model for Face Representation and Sketching," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 6, pp. 955–969, 2008 (cit. on p. 45).

[167] H. Yan, "Cost-sensitive ordinal regression for fully automatic facial beauty assessment," *Neurocomputing*, vol. 129, pp. 334–342, 2014 (cit. on p. 76).

[168] S. Yan, H. Wang, X. Tang, and T. S. Huang, "Learning Auto-Structured Regressor from Uncertain Nonnegative Labels," in *IEEE International Conference on Computer Vision*, 2007 (cit. on pp. 45, 56).

[169] S. Yan, X. Zhou, M. Liu, M. Hasegawa-Johnson, and T. S. Huang, "Regression from patch-kernel," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2008 (cit. on p. 45).

[170] J. Yang, J. Wright, T. Huang, and Y. Ma, "Image super-resolution as sparse representation of raw image patches," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2008 (cit. on p. 13).

[171] X. Yang, B.-B. Gao, C. Xing, Z.-W. Huo, X.-S. Wei, Y. Zhou, J. Wu, and X. Geng, "Deep Label Distribution Learning for Apparent Age Estimation," in *IEEE International Conference on Computer Vision Workshops*, 2015 (cit. on pp. 47, 48, 63).

[172] Y. Yang, N. P. Galatsanos, and A. K. Katsaggelos, "Projection-based spatially adaptive reconstruction of block-transform compressed images," *IEEE Transactions on Image Processing*, vol. 4, no. 7, pp. 896–908, 1995 (cit. on p. 9).

[173] Z. Yang and H. Ai, "Demographic Classification with Local Binary Patterns," in *International Conference on Biometrics*, 2007 (cit. on p. 45).

[174] J. Yao, S. Fidler, and R. Urtasun, "Describing the scene as a whole: Joint object detection, scene classification and semantic segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2012 (cit. on p. 23).

[175] D. Yi, Z. Lei, and S. Z. Li, "Age Estimation by Multi-scale Convolutional Network," in *Asian Conference on Computer Vision*, 2014 (cit. on pp. 45, 64).

[176] W. Youyou, M. Kosinski, and D. Stillwell, "Computer-based personality judgments are more accurate than those made by humans," *National Academy of Sciences*, vol. 112, no. 4, pp. 1036–1040, 2015 (cit. on p. 77).

[177] C.-N. J. Yu and T. Joachims, "Learning structural SVMs with latent variables," in *International Conference on Machine Learning*, 2009 (cit. on pp. 24, 29).

[178] A. L. Yuille and A. Rangarajan, "The concave-convex procedure," *Neural Computation*, vol. 15, no. 4, pp. 915–936, 2003 (cit. on p. 29).

[179] M. D. Zeiler and R. Fergus, "Visualizing and Understanding Convolutional Networks," in *IEEE European Conference on Computer Vision*, 2014 (cit. on pp. 67, 69, 99, 119).

[180] R. Zeyde, M. Elad, and M. Protter, "On single image scale-up using sparse-representations," in *International conference on curves and surfaces*, 2012 (cit. on pp. 11, 12).

[181] G. Zhai, W. Lin, J. Cai, X. Yang, and W. Zhang, "Short Communication: Efficient Quadtree Based Block-shift Filtering for Deblocking and Deringing," *Journal of Visual Communication and Image Representation*, vol. 20, no. 8, pp. 595–607, 2009 (cit. on p. 10).

[182] Y. Zhang and D.-Y. Yeung, "Multi-task warped Gaussian process for personalized age estimation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2010 (cit. on pp. 45, 64, 79).

[183] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, and A. Oliva, "Learning deep features for scene recognition using places database," in *Advances in Neural Information Processing Systems*, 2014, pp. 487–495 (cit. on p. 97).

[184] Y. Zhu, Y. Li, G. Mu, and G. Guo, "A Study on Apparent Age Estimation," in *IEEE International Conference on Computer Vision Workshops*, 2015 (cit. on pp. 47, 63).

CURRICULUM VITAE

PERSONAL DATA

Name	Rasmus Rothe
Date of birth	25th December 1989
Place of birth	Bremen, Germany
Citizenship	German

EDUCATION

2013 – 2016	*ETH Zurich, Computer Vision Laboratory* Doctoral studies supervised by Prof. Luc Van Gool
2012 – 2013	*Princeton University* Oxford-Princeton Exchange
2009 – 2013	*Balliol College, University of Oxford* MEng Engineering Science
2006 – 2009	*Schulzentrum Boerdestrasse* High school

WORK EXPERIENCE

2013 – 2016	*HackZurich* Founder
2012	*Google* Associate Product Manager Intern
2011	*The Boston Consulting Group* Visiting Associate
2010	*California Institute of Technology* Research Fellow at the Department of Computation & Neural Systems
2008 – 2009	*DFKI (German Research Center for Artificial Intelligence)* Part-time research intern

AWARDS

2015	ICCV 2015, ChaLearn LAP 2015 Best Paper Award
2015	ICCV 2015, Winner of LAP apparent age estimation challenge
2013	Head of Department's Prize for excellent performance in examinations (Oxford University)
2010	Lubbock Scholar for the highest score in the Preliminary Examination in College
2010	Fellow of the A.C. Lange Foundation
2009	Fellow of the Studienstiftung des Deutschen Volkes (German National Academic Foundation)
2009	Distinction for the best male Abitur in the state of Bremen (Karl-Nix Foundation)
2008	Jugend Forscht German vice champion
2007	Robocup Junior world champion