Correspondence Estimation in Human Face and Posture Images

VAHID KAZEMI

Doctoral Thesis
Stockholm, Sweden 2014
Abstract

Many computer vision tasks such as object detection, pose estimation, and alignment are directly related to the estimation of correspondences over instances of an object class. Other tasks such as image classification and verification if not completely solved can largely benefit from correspondence estimation. This thesis presents practical approaches for tackling the correspondence estimation problem with an emphasis on deformable objects.

Different methods presented in this thesis greatly vary in details but they all use a combination of generative and discriminative modeling to estimate the correspondences from input images in an efficient manner. While the methods described in this work are generic and can be applied to any object, two classes of objects of high importance namely human body and faces are the subjects of our experimentations.

When dealing with human body, we are mostly interested in estimating a sparse set of landmarks – specifically we are interested in locating the body joints. We use pictorial structures to model the articulation of the body parts generatively and learn efficient discriminative models to localize the parts in the image. This is a common approach explored by many previous works. We further extend this hybrid approach by introducing higher order terms to deal with the double-counting problem and provide an algorithm for solving the resulting non-convex problem efficiently. In another work we explore the area of multi-view pose estimation where we have multiple calibrated cameras and we are interested in determining the pose of a person in 3D by aggregating 2D information. This is done efficiently by discretizing the 3D search space and use the 3D pictorial structures model to perform the inference.

In contrast to the human body, faces have a much more rigid structure and it is relatively easy to detect the major parts of the face such as eyes, nose and mouth, but performing dense correspondence estimation on faces under various poses and lighting conditions is still challenging. In a first work we deal with this variation by partitioning the face into multiple parts and learning separate regressors for each part. In another work we take a fully discriminative approach and learn a global regressor from image to landmarks but to deal with insufficiency of training data we augment it by a large number of synthetic images. While we have shown great performance on the standard face datasets for performing correspondence estimation, in many scenarios the RGB signal gets distorted as a result of poor lighting conditions and becomes almost unusable. This problem is addressed in another work where we explore use of depth signal for dense correspondence estimation. Here again a hybrid generative/discriminative approach is used to perform accurate correspondence estimation in real-time.
Acknowledgments

Many people have contributed directly or indirectly to this thesis. First and foremost I would like to thank my supervisors Josephine Sullivan and Stefan Carlsson. Josephine helped me in many different aspects before and during my PhD studies and without her this thesis would not have happened. Thanks to Stefan who helped broaden my perspective over the whole field of computer vision.

During my visit to Microsoft Research I had the chance to work with some of the greatest researchers in the field of computer vision and machine learning whom I would like to thank. Thanks to Pushmeet Kohli, Jonathan Taylor, Cem Keskin and Shahram Izadi for their contribution.

Thanks to my co-authors at KTH, Hossein Azizpour and Magnus Burenius with whom I enjoyed working with. Many more people have indirectly contributed to this work, I specially want to thank Hedvig Kjellström, Atsuto Maki, Carl Henrik Ek and Arne Leijon.

I also want to thank my current and past colleagues at CVAP who made the hardship of scientific research much easier. I would like to thank Omid, Oscar, Miro, Ali, Heydar, Cheng, Martin, Yuquan, Alessandro, Akshaya, Virgil, Püren, Yasemin, Florian, John, Märten, Christian, Petter, Niklas, Jeannette, Javier, Alper, Marianne, Andrzej, Dan, Alireza, Babak, Nils, Ioannis, Rasmus, Kaiyu, Johannes, Emil, Alejandro, Francisco, Johan, Erik, Marina, Rares Andrei, Fredrik, Sergio, Zhan, Michele, and David.

Special thanks to Patric Jensfelt who accepted me to his master program and continued his support until my graduation, and thanks to Danica Kragic for doing such a wonderful job single handedly managing CVAP.

Lastly I would like to thank my family who always encouraged me to pursue science. I hope you are proud of me.
List of Papers

The thesis is based on the following papers:


The paper is an extended version of the following award winning paper:


¹Magnus Burenius and I have contributed equally to this paper. I focused on the 2D aspects of the problem while Magnus focused on the 3D aspects. The paper won the award for Best Industry Paper at BMVC ’13.

²The paper is the result of my internship at Microsoft Research in Cambridge.
Part I

Introduction
Chapter 1

Introduction

Vision, like many other human abilities, is very familiar to us, as we use it all the time in our daily life, yet we know very little about the underlying process that makes us see and understand things. Computer vision is an emerging branch of computational science that aims to construct algorithms that replicate this ability on software platforms, enabling computers to see what we see and understand the world around us as we do.

At the time of writing this thesis, we are already seeing applications of computer vision in our daily life. Almost every camera comes with an automatic face detector and a feature tracker which helps the camera to keep the focus on the target. Computer vision has been successfully used in the movie industry to create realistic facial animations in movies such as James Cameron’s *Avatar*. With the advent of *Kinect*, Microsoft brought a state-of-the-art body tracking system to consumers, enabling controller-free gaming. Using computer vision technology, Google has been able to produce driver-less cars that can autonomously traverse through city traffic.

These are just a few examples of what we have achieved so far, but computer vision has much greater potential than what we have seen. As the field is moving forward, we are developing algorithms that are faster and more accurate. The consensus is that, someday, computer vision will reach and surpass human performance, and that is when we will see another technological revolution perhaps with as big an impact or greater than the invention of computers.

1 Problem Statement

This thesis, which is based on a collection of papers [22, 23, 20, 24, 19, 21], aims to present practical solutions for solving computer vision problems. We are specifically interested in the problem of correspondence estimation over instances of an object class (See figure 1). Many of the applications we mentioned earlier are in someway related to the correspondence estimation problem. We define the problem...
CHAPTER 1. INTRODUCTION

Figure 1: Given an input image of an object, like a car, the correspondence estimation problem deals with locating points on the surface of a generic object model that correspond to object’s pixels on the image. Variations in pose, lighting, color and texture make this a challenging problem in computer vision.

of correspondence estimation as follows:

Given an input image of an object, for each object’s pixel, find the corresponding location on the surface of a generic object model.

Note that the above definition is commonly referred to as dense correspondence estimation. Sometimes we are interested in finding the correspondences for a subset of these pixels. For example when dealing with humans, we are usually interested in finding the location of a small number of landmarks corresponding to body joints. For face applications we are often interested in locating landmarks corresponding to the location of eyes, mouth corners, and etc. (See figure 2). In these cases we fix the location of landmarks on the model, and the task is to find the corresponding points on the image. This is an alternative way of presenting the problem, but it is essentially the same problem.

Correspondence estimation is a challenging problem in computer vision. This is primarily because points on the surface of objects greatly vary in appearance in 2D images with small variations in pose, camera parameters, and lighting. The collection of papers in this work focus on the case of deformable objects (See figure 2). Correspondence estimation for deformable objects is even more challenging, because deformation can also change the appearance of surface points.

There are lots of applications in computer vision that in someway involve estimation of correspondences. An example, which was mentioned earlier, is human body tracking. This is the core technology behind Microsoft’s Kinect gaming.

1Throughout this thesis we use the terms landmarks or surface points interchangeably to refer to certain points on the object that we are interested in putting in correspondence.
2. DISCRIMINATIVE MODELING

Figure 2: The focus of this thesis is on the problem of estimating correspondences across images of deformable objects (particularly faces and the human body). Correspondence estimation of deformable objects is more challenging compared to rigid objects, since the appearance of surface points on the object greatly vary with deformation.

platform\cite{1}, where players are able to control games by body gestures and without the need for an external controller. We also mentioned the use of computer vision techniques in the movie industry for creating realistic animations through facial performance capture. This is another direct application of the correspondence estimation problem. There are many more applications in medical imaging for reconstruction and tracking of bones and tissues through analyzing x-ray images. Furthermore estimating correspondences is an essential part of other computer vision methods such as face recognition and action recognition, which have important applications in human-computer interaction and security.

A generalization of the correspondence estimation problem which is beyond the scope of this thesis treats all different classes of objects as a single deformable object class and aims to find inter-class as well as intra-class correspondences over images. While we did not have enough time to explore further on this idea, we find it fascinating and worth putting some thought into.

2 Discriminative Modeling

A common approach for finding correspondences across images of an object is to use discriminatively trained classifiers that can distinguish certain landmarks from the background. This for example can be achieved by extracting HOG (histogram of oriented gradients)\cite{2} features from the image patches and learning SVM (support vector machine)\cite{3} filters that can best separate image patches belonging to a certain landmark from anything else. At test time we then exhaustively evaluate all the possible patches and pick the ones with the highest response. This is a standard approach for building object detectors\cite{4}, but it can also be used for landmark localization. This approach though suffers from a number of problems.

One problem is that a feature descriptor such as HOG discards some spatial
information. This is done to gain robustness to moderate amounts of deformation, which comes at the cost of lower discriminability. An alternative approach is to directly use the RGB (or depth in case we have access to a range sensor) signal as the input feature. In fact this is the approach that we take in papers D and E. One should note though that the amount of variation in the RGB signal between different instances of a single surface point is much higher than that of HOG. That means that one needs to use a model with a much higher capacity to recognize the landmarks over multiple images.

Another problem with this approach is caused by the fact that the location of surface points are highly correlated and can not be treated as independent random variables. If we try to locate these landmarks independently we risk producing inconsistent estimations. For example in a scenario where we are interested in locating facial landmarks, using independent classifiers for each landmark might lead to confusing the left and right eye. One solution to this problem is to use a global regressor that can jointly estimate the location of landmarks (paper D). An alternative solution is to explicitly model the relations between landmarks using a generative model (paper A).

Discriminative modeling is a powerful approach that is widely used in computer vision literature [12, 34, 35, 6, 11, 10, 37] and also in this thesis, but it comes with a major shortcoming. The problem is related to the generalization ability of discriminative models. To achieve good performance on the test set, a discriminative model needs to be trained with lots of labeled examples. Insufficient number of training examples is the root of all evil when learning discriminative models. It leads to the common problem of overfitting, which occurs when the model learns relations that hold for training examples but do not generalize to test examples. There are however ways to overcome this problem. Regularization techniques which we will briefly talk about in chapter 2 can reduce the chance of overfitting. This is when we utilize additional prior information to constrain model parameters. Another solution that we exploit in papers D and E is to extend the training data by generating synthetic images using computer graphics techniques. Although, in some cases synthesizing realistic examples is too complex and we need to use real images. The labeling process often requires human supervision and thus is time consuming and undesirable. A way to address this problem is to use generative models which we describe now.

3 Generative Modeling

In a generative approach we model the distribution of the observed data. The standard approach to estimate the parameters of such a model is maximizing the likelihood of observed examples – this is commonly referred to in short as maximum likelihood [2]. One of the reasons for why generative modeling is attractive is because this approach allows for the use of unlabeled examples. This is a desirable property since most of the time in computer vision applications unlabeled examples are cheap.
A HYBRID APPROACH

and abundant. However, the generative modeling approach also comes with a set of drawbacks.

Firstly, discriminative models have been shown to outperform their generative counterparts given enough training examples [31, 27]. This observation has motivated use of discriminatively trained generative models [15]. The problem though here is that these approaches can no longer take advantage of unlabeled examples.

Another drawback with the generative modeling approach is that inferring the unknown parameters of the model is often non-trivial, expensive, and sometimes intractable. While in some certain cases efficient inference is possible, it may require over-simplification of the model [14, 33, 15, 39]. This is an issue that we will visit in paper [B]. A solution to this problem is combining discriminative models with generative models. We provide some examples for the use of this approach in the next sections.

4 A Hybrid Approach

The vast majority of papers in this thesis as well as some previous works [7, 1, 36] use a combination of generative and discriminative modeling, this is because most of the time neither a purely generative nor discriminative model leads to satisfying results. In the following we briefly describe how a hybrid approach is used to solve correspondence estimation problem in different papers included in this thesis.

- Paper [A] uses a generative approach to model the configuration of facial parts namely the eyes, nose, and the mouth. A discriminative model is then used to regress the location of facial landmarks (e.g. corners of eyes) from the patches extracted from corresponding parts.

- Paper [B] uses a generative model to produce multiple hypotheses for the pose of a human and then uses a discriminative model to select the best configuration.

- In paper [C] a discriminative model is used to estimate the likelihood of body joints across multiple views. A generative model is then used to infer a single consistent configuration from the likelihood maps.

- Finally in [E] a discriminative model is used to estimate an initial value for hidden variables of our generative model. In an iterative procedure, unobserved parameters are then further optimized to estimate the final correspondences.

Often there is not a right or wrong way of combining generative and discriminative models, but the determining factor is the type of data.
Chapter 2

Background

The purpose of this chapter is to provide some background of the key concepts discussed in this thesis. Each section of this chapter is independent of the others and can be read or skipped if the concepts are familiar to the reader.

1 Feature Descriptors

In computer vision, a feature descriptor is referred to a transformation of the input image that is used as the input to the computer vision model. An ideal feature descriptor should be compact, specific, invariant to noise, and it should disentangle physical information, but these qualities are often contradictory and compromises have to be made when designing features. In the following, we discuss these properties in more detail.

An ideal feature descriptor should be compact. If the same information can be represented in fewer number of dimensions, often we prefer the low dimensional representation. A high dimensional feature space is not desirable because it usually adds to the computational time of our algorithms, and more importantly can lead to the problem of curse of dimensionality [18].

Feature descriptors also should be specific while invariant to noise. This is a rather subjective trade-off. We want the feature to be specific so that it can distinguish between different attributes that are relevant to the recognition task, yet we do not want the descriptor to be sensitive to noise and irrelevant attributes. For example in a human pose estimation application, we want the feature to be sensitive to the pose, but invariant to the clothing of the person. Obviously, such a property is task specific, a more generic desirable property is disentanglement of physical information. In other words, an ideal feature descriptor should separate physical properties of the object(s) in the image.

The next obvious question is, how do we design such features? The traditional approach is to hand engineer the features, i.e. use our prior knowledge about the images to design the features. An alternative approach which is beyond the scope
of this thesis is to try to learn these features with tons of examples.

In this section we suffice to describe a well known handcrafted generic feature
that has been empirically shown to perform well for a variety of visual recognition
tasks.

1.1 Histogram of Oriented Gradients

The HOG feature was introduced by Dalal and Triggs [9], and has been amongst the
best performing generic feature descriptors. As its name implies, the HOG descrip-
tor consists of a histogram of gradient orientations. The procedure for calculating
this descriptor is as follows.

Starting from an input image, the image is divided to an array of equally sized
regions called cells. For each of these cells, a histogram of image gradients is calcu-
lated where each bin of the histogram corresponds to a certain gradient orientation.
Furthermore contiguous cells are grouped together to form blocks. The histogram
 corresponding to each cell is then normalized with respect to all the nearby cells
within its block to gain some invariance to global lighting.

The HOG feature has been shown to work very well particularly in conjunction
with linear SVM classifiers [9], and until recently the state-of-the-art object detection
methods [15] used HOG as the input feature. In this thesis we also extensively
use HOG features as the input to classifiers for detecting face and body parts (paper
A, B, and C).

2 Linear Models

This section will introduce two popular linear models that are widely used in com-
puter vision and other pattern recognition applications. We start off by introducing
the ridge regression method, which is a regularized least-square model, and is used
to approximate real valued functions. After that, we briefly talk about the classi-
fication problem and how an optimal hyperplane can be estimated to separate two
classes using support vector machines (SVM).

2.1 Ridge Regression

In a visual recognition task we often want to estimate a set of labels \( y \in \mathbb{R} \) from
a representation of the input image \( x \in \mathbb{R}^D \) where \( D \) is the feature dimension.
We often assume that we can find a function mapping \( f \) such that

\[
y = f(x).
\]  

(2.1)

A particular class of functions to express this mapping is the class of linear
models.

\[
y = w \cdot x
\]  

(2.2)
A common approach to learn the parameters ($w$) of such a model from a set of training examples $\{(x_1, y_1), \ldots, (x_N, y_N)\}$ is by minimizing the least square error

$$\hat{w} = \arg \min_w \sum_{i=1}^N ||w \cdot x_i - y_i||^2.$$  \hspace{1cm} (2.3)

Unfortunately such a naive approach to parameter estimation often leads to very bad results in practice. The problem is that in many cases, we do not have enough examples to estimate the model parameters, making the optimization an ill-posed problem. Even when the number of examples is higher than the number of unknown parameters, we still run the risk of overfitting to the training data. The problem arises when we have noisy measurements and/or the model is too flexible. In such cases it is necessary to regularize the model by introducing a shrinkage term

$$\hat{w} = \arg \min_w \sum_{i=1}^N ||w \cdot x_i - y_i||^2 + \lambda ||w||^2,$$  \hspace{1cm} (2.4)

where $\lambda \geq 0$ is the complexity parameter that is inversely proportional to the degrees of freedom of the model. This approach is commonly referred to as ridge regression\[18\], and has a simple closed form solution

$$\hat{w} = (X^T X + \lambda I)^{-1} X^T y,$$  \hspace{1cm} (2.5)

where $X = [x_1, \ldots, x_N]^T$ is a matrix consisted of training features, and $y = [y_1, \ldots, y_N]^T$ is a vector of the corresponding labels.

### 2.2 Support Vector Machines

So far we have talked about solving general regression problems with linear models where we have a continuous target space. In classification problems however the target space is discrete and finite. For example, in a typical object classification task, we are interested in determining the object category corresponding to an input image from a limited set of possibilities (e.g. human, bird, cat, etc.). Classification problems are an important and well studied subject in machine learning.

For a special case of classification problems where we have binary labels $y \in \{-1, 1\}$, Vapnik \[38\] suggests a method for finding the optimal hyperplane that separates the two classes. This method, commonly referred to as support vector machines (SVM), defines the optimal hyperplane as the hyperplane that perfectly separates two classes with the maximum possible margin. The data samples that lie on the margin are called the support vectors. Such a hyperplane can be found by optimizing the following objective function (\[18\])

$$\hat{w} = \arg \min_{w, b} ||w||^2$$  \hspace{1cm} (2.6)

subject to $\forall i, \ y_i(w \cdot x_i + b) \geq 1,$  \hspace{1cm} (2.7)
CHAPTER 2. BACKGROUND

where $b$ is the bias term. This optimization problem turns out to be convex, and standard quadratic programming techniques can be used to solve this problem.

Note that so far we assumed that the data is linearly separable. For cases where data is not linearly separable, we use a variant of SVM classifier called soft-margin SVM which allows for some outliers to appear on the wrong side of the margin. This is achieved by introducing a set of slack variables $\{\xi_1, ..., \xi_N\}$, which correspond to the amount which each sample violates the margin, and altering the optimization problem as follows

$$\hat{w} = \arg \min_{w, \xi} ||w||^2$$  \hspace{1cm} (2.8)

subject to $\forall i, \ y_i(w \cdot x_i + b) \geq 1 - \xi_i$, and $\xi_i \geq 0$. \hspace{1cm} (2.9)

At test time, we simply evaluate the svm model as follows

$$y = \text{sign}(w \cdot x + b).$$ \hspace{1cm} (2.11)

3 Tree Based Models

In the previous section we described two common and widely used linear methods for solving regression and classification problems. These models, however, have limited flexibility, which in many cases is not enough for modeling complex relations. There is a whole class of methods that deal with modeling nonlinear relations. But here we limit ourselves to the discussion of tree based methods as they are most used in this thesis. Methods presented in this section are based on the concept of decision trees. Decision trees are simple, intuitive, and efficient models that can be used for solving a variety of classification and regression tasks.

Decision trees consist of split nodes and leaves. The most common type of decision tree is the binary type, where decisions at each split node are based on binary tests, which are functions of the input feature. The result of these tests (whether it is true or false) determine the next node to visit (either right or left). Each leaf usually returns a single label corresponding to the most probable label which is determined based on the statistics of the training examples reaching that leaf. Starting from the root node, the decision tree is traversed until reaching a leaf node, the output of the decision tree is then simply the value which is stored at that leaf.

In computer vision, we often need to model very complex relations between the input features and the labels, and therefore it is often impossible to design a decision tree by hand. Instead, we use learning algorithms to build the tree automatically. CART (short for classification and regression trees) \cite{1} is an example of such an algorithm, which we describe briefly next.

Assuming that we have access to a set $Q$ of pairs of labeled examples $Q = \{(x_1, y_1), ..., (x_N, y_N)\}$. We start by building a pool of binary tests that split the
3. TREE BASED MODELS

feature space into two partitions \((Q_l, Q_r)\), and then select the split which maximizes the information gain \((\mathcal{I}G)\) defined as follows

\[
\mathcal{I}G = \mathcal{H}(Q) - \sum_{s \in \{l,r\}} |Q_s| \mathcal{H}(Q_s),
\]

(2.12)

where \(\mathcal{H}(Q)\) represents the information entropy. For classification problems where we have a discrete set of labels \(y \in \{1, ..., C\}\), we can define the entropy as

\[
\mathcal{H}(Q) = -\sum_{y=1}^{C} P_Q(y) \log P_Q(y),
\]

(2.13)

where \(P_Q(y)\) corresponds to the ratio of label \(y\) in \(Q\)

\[
P_Q(y) = \frac{1}{|Q|} \sum_{i \in Q} \mathbb{I}(y_i = y).
\]

(2.14)

where \(\mathbb{I}\) is the indicator function. This procedure is repeated recursively for each node until we can no longer increase the information gain. At each leaf then we store the distribution of labels \((P_Q)\) from the training examples that reached that node.

3.1 Ensemble Methods

Decision trees are rarely used as a standalone machine learning tool to solve computer vision problems because of their poor generalization ability. Instead, decision trees are often used as a building block for ensemble methods. Ensemble methods combine multiple weak models to create a stronger model with higher predictive power. One should note though that the improvement can only be achieved if the base learners are diverse. In other words, it is crucial that the weak learners do not make the same mistakes. This can be done in various ways.

Bagging is a common technique to ensure diverse base models in an ensemble. The idea here is to split the training data into multiple subsets and build one base model for each part of data. Another technique, which is commonly used in conjunction with tree-base models, is to introduce randomness in feature selection. This can be achieved for example by reducing the pool of features during training of decision trees.

Both of the ideas mentioned above are used in random forests \([4]\). After building \(K\) decision trees \((T_1, ..., T_K)\) in this way, the outputs are simply averaged to produce the final prediction of the random forest \(f\).

\[
f(x) = \frac{1}{K} \sum_{k=1}^{K} T_k(x)
\]

(2.15)
In paper C, we describe how a random forest model can be used to classify pixels of an image of a football player to different body parts or background. An alternative strategy to diversify the base models is through boosting. AdaBoost [17] is an example of boosting method that incrementally builds an ensemble from a set of weak classifiers. At the first stage, the ensemble is initialized with a base model that is slightly better than random guessing. At each later stage, the algorithm re-weights the examples based on the prediction error of the ensemble, and then trains and adds (to the ensemble) a new base model that focuses on the training examples that are not well explained by the ensemble.

Gradient tree boosting algorithm, introduced by Hastie et al. [18], is an alternative boosting method that can be used to solve both classification and regression problems. An overview of this algorithm is provided in paper D, where we use the gradient boosting algorithm to learn a global regression from an input face image to facial landmarks.

4 Pictorial Structures

For rigid objects, the pose of an object can simply be represented by a similarity matrix, including a translation and rotation. For non-rigid objects however we need a more flexible representation. This is often achieved by representing the pose with a set of landmarks. For example in case of human body, we commonly define these landmarks over the major body joints. The problem of pose estimation is then reduced to locating these landmarks in the image.

In the last chapter we gave an example of how such a problem can be solved by learning classifiers that can identify each landmark independently. We also discussed the flaws of such a naive approach. Mainly the fact that the location of these landmarks are highly correlated and any independence assumption might produce inconsistent estimations of landmarks leading to invalid poses. One solution to this problem is to explicitly model the relation between these landmarks with a generative shape model. One such model, commonly referred to as pictorial structure, was introduced by Fischler and Elschlager ([16]) and later developed by Felzenszwalb et al. ([14, 15]) for the task of pose estimation and object detection.

A pictorial structure is a constellation of moving parts. In this model, each part has its own independent appearance model that estimates the likelihood of a part for each pixel on the image. The configuration of the parts is constrained by pairwise spatial constraints. This model can be best expressed with a graph structure, $G = (V, E)$ where each vertex $v_i \in V$ corresponds to a part, and each edge $(v_i, v_j) \in E$ corresponds to a connection between two parts. Let $p_i$ be the coordinate of the center of $i$th part, then the pose of the object can be represented by a vector $p = (p_1, ..., p_K)$ where $K = |V|$ is the number of parts. Assuming that we have a function $s_a(p_i)$ that calculates the likelihood of $i$th part, and a deformation function $s_d(p_i, p_j)$ that assigns a likelihood to the configuration of each pair $v_i$ and $v_j$, the pictorial structure model then assigns a global score to the configuration of
4. **PICTORIAL STRUCTURES**

parts as follows

\[ S(p) = \sum_{v_i \in V} s_a(p_i) + \sum_{(v_i, v_j) \in E} s_d(p_i, p_j). \]  

(2.16)

This function can then be maximized to find the optimal configuration of parts

\[ p^* = \arg \max_p S(p). \]  

(2.17)

Felzenszwalb et al. [14] show that if we limit the connections of this graph to a tree structure, and use quadratic functions to model the deformation \( s_d \), we can then solve the inference problem efficiently using generalized distance transforms [14]. This discovery in conjunction with a later paper on discriminative learning of pictorial structure model parameters [15] revolutionized the field of object detection and pose estimation and until recently pictorial structure based models were the state of the art in almost all the standard general object detection benchmarks such as PASCAL VOC [13]. We also extensively use this model throughout this thesis. In particular paper [B] and [C] use a pictorial structure model for human pose estimation, and [A] applies this model to tackle the problem of face alignment.
Chapter 3

Summary of papers

A  Face Alignment with Part-Based Modeling

This paper addresses the problem of face alignment, that is given an image of a face we want to localize a set of landmarks on the image. These landmarks are defined over the boundaries of the eyes, mouth, and the nose as is shown in figure 1. The landmarks are chosen to capture the major deformations of the face, and therefore can be used for a variety of applications including facial expressing tracking and identity recognition.

Our aim in this paper is to learn a mapping from an image to the landmarks. We know though that this global mapping is highly nonlinear and therefore learning

![Figure 1: This figure shows the benefit of using parts in the performance of a regression function to accurately predict the location of landmark points. In both cases a linear regression model is learnt to map the appearance descriptors inside the patches to the location of the landmarks associated with the patch. Green lines represent the ground truth shape and the red lines are the prediction of the regression function. As can be seen greater accuracy is achieved when (a) using a separate regression function for each localized part as opposed to (b) one regression function from the global face patch.](image-url)
this function requires lots of training data that we assume we do not have access to. Instead, our strategy in this paper is to learn multiple simpler regressors for each individual facial part (i.e. eyes, nose, and the mouth) that can independently regress the location of corresponding landmarks. These individual partial models can be trained with much less examples, since the variation in appearance of local parts are much lower compared to that of the whole face. In fact we show that even linear regressors suffice to model the mapping in the examined datasets. Figure 1 shows how such a part based approach performs compared to a global regression approach.

Note though that this approach requires us to have a good estimate of the location of major facial parts. This is done by utilizing a pictorial structure model. We model the appearance of individual parts with a multivariate Gaussian distribution, and use a simple star model (with the nose at the root, and eyes and the mouth as leaves) to model the deformation of the parts.
B. USING RICHER MODELS FOR ARTICULATED POSE ESTIMATION OF FOOTBALLERS

B Using Richer Models for Articulated Pose Estimation of Footballers

In chapter 2, we briefly discussed pictorial structure models. A number of limitations are enforced on pictorial structure models to make the inference tractable (for example, we are limited to pairwise quadratic deformation functions and the dependency graph defined over parts can not contain a loop[14]). These limitations have a direct effect on the performance of pictorial structure models. In other words, the maximum scored pose using a pictorial structure model in many cases is not the true pose of the object. Note that this problem is not limited to pictorial structures, this in fact is a general problem with generative modeling approach that often accurate modeling leads to intractable inference. Our solution to such a problem is very simple but we show that it can lead to significantly better performance in pose estimation problems. It is based on the observation that, although the global optimum of the pictorial structure model in many cases is not the true configuration, but nevertheless the true configuration consistently gets a high score.

Figure 3: Given an image of a football player, we want to determine a set of landmarks representing his pose. This is done by using a pictorial structure based model to generate multiple candidates for the pose and selecting the best one using a more accurate model.

Assuming that we have access to two models, first a simple model that is fast to evaluate but not very accurate, and second an accurate model that is too expensive to exhaustively evaluate for all the possible configurations. Our solution is then to use the simple model to generate a set of highly likely candidates, and then only evaluate the expensive model on these configurations to pick the best one.

Generating multiple hypotheses might not be straightforward in a general case. The problem here is that the model might rank very similar poses as the top scoring poses. In these cases, we might need to introduce additional constraints to enforce diverse solutions[32]. Note that producing diverse solutions is essential to quickly explore a large portion of the parameter space.

Our more accurate model adds two additional components to a state of the art pictorial structure based model[39]. Firstly, we enforce an exclusion principle to avoid the problem of double counting of the limbs, which is a common problem.
with pictorial structure based models. In other words, our model doesn’t assume
that the location of the left and right limbs such as arms and legs are independent
random variables. Furthermore our model includes a segmentation score that scores
configurations that explain more foreground pixels higher. Inferring the most likely
pose with respect to such a complex model is very expensive. But we use the
framework of \cite{32} to generate multiple hypotheses for the pose and only evaluate
our model on the 1000 top scored configurations to pick the best one. Figure 4 shows
an example where our reranking model is able to improve the result of a pictorial
structure base model. On average, we show that the top scoring configuration
returned by our model is 15\% more likely to be the true configuration.

Figure 4: This figure shows (a) the result of FMP compared to (b) our reranking function,
in addition to (c) the results of picking the closest configuration to the ground truth from
a set top 1000 configurations.
C. Multi-view Pose Estimation of Human Body

This paper focuses on the problem of multi-view pose estimation. Given images of a football player captured with multiple calibrated cameras, our goal is to reconstruct the pose of the body in 3D (See figure 5). Our strategy here is to use an efficient discriminative model to estimate the likelihoods of each body part over a 3D voxel, and then use a generative model based on 3D pictorial structures to produce a consistent hypothesis for the pose of the person across multiple views.

Figure 5: A general overview of our multi-view pose estimation framework. A 2D discriminative model is first used to classify pixels in each image as belonging to a part or the background. The results are then back-projected to a 3D volume. We find corresponding mirror symmetric parts across views by introducing a latent variable. Finally, a part-based model is used to estimate the 3D pose.
Figure 6: Final 3D poses obtained by taking, for each part independently, its most probable state over the grid. The mirror ambiguity is solved jointly. Estimation is red and ground truth is blue.

To learn a discriminative model that can directly estimate the likelihood of each part in 3D, one needs access to labeled 3D data and the associated calibrated cameras. There are two major problems with this approach, firstly collecting 3D data and labeling them is very expensive, and furthermore this approach requires fixed camera views – 3D data captured from a certain camera setup can only be used for that particular setup, and a small change in the pose of any of the cameras would require recollection of the training data.

In contrast, our approach relies on 2D discriminative models that assign a likelihood to each pixel from each view independently, and we then project these scores on a discretized 3D space to produce 3D likelihoods. At the final stage a 3D pictorial structure model is used to select the optimal configuration of body parts with respect to simple limb length priors. Figure 6 shows examples of the final 3D pose estimated by our model comparing to the ground-truth annotations.
D. ONE MILLISECOND FACE ALIGNMENT WITH AN ENSEMBLE OF
REGRESSION TREES

D One Millisecond Face Alignment with an Ensemble of Regression Trees

In this paper we again revisit the problem of face alignment, but with a completely
different strategy. In section A we briefly talked about the challenges of a global
regression approach for estimating facial landmarks, namely the fact that the global
mapping between the image features and the landmarks is highly nonlinear and dif-
ficult to model, and building such a model would require lots of training examples.
Instead of averting the problem as we did in A, here, we choose to address the prob-
lem, and we show that such an approach can achieve state of the art performance for
facial landmark detection while being much faster than any other previous method.
(See figure 7)

Figure 7: Selected results on the HELEN dataset. An ensemble of randomized
regression trees is used to detect 194 landmarks on face from a single image in a
few milliseconds.

Our aim here is to learn a global regression model that directly estimates the
location of landmarks from the input image. This is achieved by using an ensemble
model consisting of thousands of shallow regression trees. To train such an ensemble
we only use 2000 face images, but the training data is extended by a factor of 20 by
warping images with random face shapes. The procedure for training the ensemble
is based on the gradient boosting algorithm, which allows for quick reduction of
training error by learning complementary decision trees. The gradient boosting
algorithm though is prone to over-fitting problem and in the paper we discuss
different strategies to avoid this problem. Furthermore we empirically show that
use of a cascade of smaller regressors is much more effective than using a single large regression model.

Figure 8 shows a few examples of the output of our method on random images from the HELEN database.
E. REAL-TIME FACE RECONSTRUCTION FROM A SINGLE DEPTH IMAGE

E Real-time Face Reconstruction from a Single Depth Image

This paper presents a novel approach for dense correspondence estimation of deformable objects from single depth images. While our method is generic and can be applied to any deformable object, our experiments are limited to human faces. Final correspondences estimated by our method are highly accurate and can be used for a variety of applications, including 3D face shape and expression reconstruction, texture unwrapping, retexturing and retargeting in real-time. (See figure 9)

Figure 9: Our method starts with estimating dense correspondences on an input depth image, using a discriminative model. A generative model parametrized by blend shapes is then utilized to further refine these correspondences. The final correspondence field is used for per-frame 3D face shape and expression reconstruction, allowing for texture unwrapping, retexturing or retargeting in real-time.

We start off by defining a generative model based on blend-shape deformations. We represent the face with a mesh model consisted of $M = 11211$ vertices. Each blend-shape then contains a $3 \times M$ dimensional delta vector (each of these blend-shapes can correspond to a particular face shape or expression). Our generative model is simply defined by a linear combination of these blend-shapes transformed by a similarity matrix that corresponds to the pose of the head.

We discuss in the paper that trying to directly minimize the parameters of such a generative model based on the reconstruction error leads to very poor results. This is because the error function is highly nonlinear and very hard to optimize. Instead, we utilize a variant of ICP algorithm. Starting with an initial estimate of the correspondences, we estimate the model parameters, and then fix the model parameters and recalculate the correspondences. This procedure is repeated multiple times until convergence. But the problem is not solved yet, such an iterative procedure still requires a good initialization of parameters to avoid divergence. This problem is addressed by using a discriminatively trained regressor that directly estimates the correspondences from the input image. This initial estimate is very crude, but after a few iterations of the ICP procedure we are able to significantly reduce the error. At a final step, we use the model parameters estimated by the ICP procedure to initialize the particle swarm optimizer (PSO) [25]. PSO then explores the nearby solutions and pick the the best configuration according to the true reconstruction error.
Figure 10: Qualitative results on real data captured using Kinect camera. From left to right, we show the input depth data, the depth data overlaid on the reconstructed model, the reconstructed model and the parts overlaid on the rgb image (which was only used for visualization). Some of these examples are from [5].

Figure 10 shows examples of the output of our method used for facial reconstruction and retexturing.
Chapter 4

Conclusion

As promised in the introduction, this thesis delivered practical solutions for some computer vision problems particularly around the subject of correspondence estimation. We would like to emphasize on the practicability of our solutions, which separates this work from many others. Many theoretically plausible solutions fail in practice because of a number of problems:

- In practice we often do not have access to infinite amount of training data and the labeled data is often very scarce.
- A particular kind of annotation might be easier to do by human labor than others.
- In practice we do not have perfect sensors and our measurements are often very noisy.
- We care about the training and test time in practice and our computational resources are limited.
- Often, our algorithms need to run on available consumer hardwares. The algorithms that can utilize the available hardware (multi-core CPUs, GPUs, and etc.) more efficiently are preferable.

These are a few examples of the limitations and concerns we face in practice, which must be taken into consideration when designing computer vision algorithms. All of the solutions provided in this thesis in one way or another are affected by these limitations, and in many cases they explain why we made the design choices and decisions we made throughout this thesis.

If we have access to a large dataset of labeled data, it makes sense to opt for a purely discriminative approach. The choice of features could potentially have a large impact on the performance of the method, but even if do not have any
But how much training data is enough to learn such mapping? Of course the answer depends on the desired level of accuracy and also variation in the data. We will need more training examples to achieve higher accuracy. Also, high variance in the data often means that the relation between the input features and the labels is more complex. A more complex relation requires a more complex model, and consequently more examples are needed to learn the mapping.

When labeled data is scarce, a generative approach might be more appropriate. We can handcraft a generative model using our prior knowledge, or learn it entirely from the data. But the common approach is to use our prior knowledge to choose a proper statistical model, and learn the parameters of the model from the data. In the recent years, more researchers have shifted their focus from handcrafted methods to data-driven methods. The motivation behind this shift is twofold. Firstly, we are often interested in generic approaches that can be applied to many different problems. Data driven approaches are usually more generic and therefore more desirable in this respect. Furthermore, we can often make a more accurate model by increasing the number of training data.

We described pros and cons of both generative and discriminative approaches in the introduction, and discussed some practical solutions to these problems throughout the thesis. While the focus of this thesis has been on the subject of correspondence estimation, some of the ideas discussed in this work can be applied to many other applications even outside the computer vision domain, and our hope is that the thesis can be useful for a broader audience who have an interest in using machine learning tools to solve real-world problems.

1 Future Work

There are a number of subjects that we did not explore in this thesis, some of which we find interesting to investigate for a future work. One important subject is representation learning. While in this work we only used predefined features, a large body of papers have been recently published about learning feature representations from data particularly using convolutional neural networks (CNNs). This line of research was sparked by the impressive results reported by Krizhevsky et al. [25] on ImageNet classification challenge. While most of the research in this area has been focused on learning pose invariant features, we find it interesting to learn pose sensitive representations. A closely related idea that we find interesting to explore is about learning inter-class pose representations. We believe it is possible to learn generic pose features across multiple classes of objects, and such a representation can potentially help improve the performance of various recognition tasks.
Bibliography


