Robust Discriminative Response Map Fitting with Constrained Local Models

Akshay Asthana[1] Stefanos Zafeiriou[1] Shiyang Cheng[1] Maja Pantic[1,2]
[1]Department of Computing, Imperial College London, United Kingdom
[2]EEMCS, University of Twente, Netherlands
{a.asthana, s.zafeiriou, shiyang.cheng11, m.pantic}@imperial.ac.uk

Abstract

We present a novel discriminative regression based approach for the Constrained Local Models (CLMs) framework, referred to as the Discriminative Response Map Fitting (DRMF) method, which shows impressive performance in the generic face fitting scenario. The motivation behind this approach is that, unlike the holistic texture based features used in the discriminative AAM approaches, the response map can be represented by a small set of parameters and these parameters can be very efficiently used for reconstructing unseen response maps. Furthermore, we show that by adopting very simple off-the-shelf regression techniques, it is possible to learn robust functions from response maps to the shape parameters updates. The experiments, conducted on Multi-PIE, XM2VTS and LFPW database, show that the proposed DRMF method outperforms state-of-the-art algorithms for the task of generic face fitting. Moreover, the DRMF method is computationally very efficient and is real-time capable. The current MATLAB implementation takes 1 second per image. To facilitate future comparisons, we release the MATLAB code[1] and the pre-trained models for research purposes.

1. Introduction

The problem of registering and tracking a non-rigid object that has great variation in shape and appearance (for example, human face) is a difficult problem and decades of research on this problem has produced a number of efficient and accurate solutions. These include commonly used methods such as Active Shape Models (ASM) [10], Active Appearance Models (AAM) [13] and Constrained Local Models (CLM) [11, 23]. Baker et al. [4] proposed several generative AAM fitting methods, some capable of real-time face tracking [17], making AAM one of the most commonly used face tracking method. However, these methods have been shown to rely heavily on accurate initialization [3]. As an alternative, several *discriminative fitting methods* for AAM were proposed [16, 20, 21, 22] that utilized the available training data for learning the fitting update model and showed robustness against poor initialization. However, the overall performance of these discriminative fitting methods have been shown to deteriorate significantly for cross-database experiments [22].

This problem has been addressed to an extent by the Constrained Local Model (CLM) framework proposed by Cristinacce et al. [11], which was later extended in the seminal work of Saragih et al. [23] who proposed a fitting method, known as the Regularized Landmark Mean-Shift (RLMS), which outperformed AAM in terms of landmark localization accuracy and is considered to be among the state-of-the-art methods for the generic face fitting scenario. However, the discriminative regression-based fitting approaches have not received much attention in the CLM framework, and hence, are the main focus of our work. As our main contribution, we propose a novel *Discriminative Response Map Fitting (DRMF) method* for the CLM framework that outperforms both the RLMS fitting method [23] and the tree-based method [26]. Moreover, we show that the robust HOG feature [12] based patch experts can significantly boost the fitting performance and robustness of the CLM framework. We show that the multi-view HOG-CLM framework, which uses the RLMS fitting method [23], also outperforms the recently proposed tree-based method [26].

We conduct experiments in controlled and uncontrolled settings. For controlled settings, we conduct identity, pose, illumination and expression invariant experiments on Multi-PIE [14] and XM2VTS [19] databases. For uncontrolled settings, we conduct experiments on LFPW [6] database. Finally, we release the MATLAB code[1] for the multi-view HOG-CLM framework with the DRMF method and the pre-trained models for research purposes. The current MATLAB implementation takes 1 second per image on an Intel Xeon 3.80 GHz processor.

2. The Problem

The aim of a facial deformable model is to infer from an image the facial shape (2D or 3D, sparse [9, 5] or dense [7]), controlled by a set of parameters. Facial deformable models can be roughly divided into two main categories:

[1]http://ibug.doc.ic.ac.uk/resources.

(a) Holistic Models that use the holistic texture-based facial representations; and (b) Part Based Models that use the local image patches around the landmark points. Notable examples of the first category are AAMs [9, 5, 25] and 3D deformable models [7]. While the second category includes models such as Active Shape Models (ASMs) [10], Constrained Local Models (CLMs) [23] and the tree-based pictorial structures [26].

2.1. Holistic Models

Holistic models employ a shape model, typically learned by annotating n fiducial points $\mathbf{x}_j = [x_j, y_j]^T{}_{j=1}^n$ and, then, concatenating them into a vector $\mathbf{s} = [\mathbf{x}_1, \dots, \mathbf{x}_n]^T$. A statistical shape model $\mathcal{S}$ can be learned from a set of training points by applying PCA. Another common characteristic of holistic models is the motion model, which is defined using a warping function $\mathcal{W}(\mathbf{x}; \mathbf{s})$. The motion model defines how, given a shape, the image should be warped into a canonical reference frame (usually defined by the mean shape). This procedure is called shape-normalization and produces shape-free textures. Popular motion models include piece-wise affine and Thin-Plate Splines [5, 2].

The holistic models can be further divided according to the way the fitted strategy is designed. In *generative holistic models* [4, 17], a texture model is also defined besides the shape and motion models. The fitting is performed by an analysis-by-synthesis loop, where, based on the current parameters of the model, an image is rendered. The parameters are updated according to the residual difference between the test image and the rendered one. In probabilistic terms, these models attempt to update the required parameters by maximizing the probability of the test sample being constructed by the model. In *discriminative holistic models*, the parameters of the model are estimated by either maximizing the classification score of the warped test image, so that it belongs to the class of the shape-free textures [16], or by finding a set of functions that map the holistic texture features to the shape model parameters [20, 21].

Drawbacks of Holistic Models: **(1)** For the case of the generative holistic models, the task of defining a linear statistical model for the texture that explains the variations due to changes in identity, expressions, pose and illumination is not an easy task. **(2)** Similarly, due to the numerous variations of facial texture, it is not easy to perform regression from texture features to shape parameters (in a recent methodology whole shape regression is performed from randomly selected texture samples [8], but unfortunately details of how this is performed were not provided in the paper). **(3)** Partial occlusions cannot be easily handled. **(4)** The incorporation of a 3D shape model is not easy due to the need of defining a warping function for the whole image; inclusion of a 3D shape model can be performed by sacrificing efficiency [1] (there is not an inverse compositional framework for the 3D case [18]) or by carefully incorporating extra terms in the cost function (which again is not a trivial task [18]).

2.2. Part Based Models

The main advantages of the part-based models are (1) partial occlusions can be easier to handled since we are interested only in facial parts, (2) the incorporation of a 3D facial shape is now straightforward since there is no warping image function to be estimated. In general, in part-based representations the model setup is $M = \{\mathcal{S}, \mathcal{D}\}$ where $\mathcal{D}$ is a set of detectors of the various facial parts (each part corresponds to a fiducial point of the shape model $\mathcal{S}$). There are many different ways to construct part-based models [23, 26], however in this paper, we will focus only on ASMs and CLMs [23].

The 3D shape model of CLMs can be described as:

$$s(\mathbf{p}) = s\mathbf{R}(\mathbf{s}_0 + \mathbf{\Phi}_s \mathbf{q}) + \mathbf{t}, \tag{1}$$

where $\mathbf{R}$ (computed via pitch r_x, yaw r_y and roll r_z), s and $\mathbf{t} = [t_x; t_y; 0]$ control the rigid 3D rotation, scale and translations respectively, while $\mathbf{q}$ controls the non-rigid variations of the shape. Therefore the parameters of the shape model are $\mathbf{p} = [s, r_x, r_y, r_z, t_x, t_y, \mathbf{q}]$. Furthermore, $\mathcal{D}$ is a set of linear classifiers for detection of n parts of the face and is represented as $\mathcal{D} = \{\mathbf{w}_i, b_i\}_{i=1}^n$, where $\mathbf{w}_i, b_i$ is the linear detector for the i^{th} part of the face (e.g., eye-corner detector). These detectors are used to define probability maps for the i^{th} part and for a given location $\mathbf{x}$ of an image $\mathcal{I}$ being correctly located ($l_i = 1$) as:

$$p(l_i = 1 \mid \mathbf{x}, \mathcal{I}) = \frac{1}{1 + e^{\{l_i(w_i^T \mathbf{f}(\mathbf{x}; \mathcal{I}) + b_i)\}}}. \tag{2}$$

where $\mathbf{f}(\mathbf{x}; \mathcal{I})$ is the feature extracted from the patch in image $\mathcal{I}$ centered at $\mathbf{x}_i$. The probability of not being correctly spotted at $\mathbf{x}$ is simply $p(l_i = -1 \mid \mathbf{x}, \mathcal{I}) = 1 - p(l_i = 1 \mid \mathbf{x}, \mathcal{I})$.

In ASM and CLMs, the objective is to create a shape model from the parameters $\mathbf{p}$ such that the positions of the created model on the image correspond to well-aligned parts. In probabilistic terms, we want to find the shape $\mathbf{s}(\mathbf{p})$ by solving the following:

$$\begin{aligned} \mathbf{p} &= \arg\max p(\mathbf{s}(\mathbf{p}) \mid \{l_i = 1\}_{i=1}^n, \mathcal{I}) \\ &= \arg\max p(\mathbf{p}) \;\; p(\{l_i = 1\}_{i=1}^n \mid \mathbf{s}(\mathbf{p}), \mathcal{I}) \\ &= \arg\max p(\mathbf{p}) \prod_{i=1}^n p(l_i = 1 \mid \mathbf{x}_i(\mathbf{p}), \mathcal{I}). \end{aligned} \tag{3}$$

In [23], by assuming a homoscedastic isotropic Gaussian kernel density estimate in a set of fixed locations $\{\mathbf{\Psi}_i\}_{i=1}^n$ for every part i, i.e. $p(l_i = 1 | \mathbf{x}_i(\mathbf{p}), I) = \prod_{i=1}^n \sum_{\mathbf{y}_i \in \mathbf{\Psi}_i} p(l_i = 1 \mid \mathbf{y}_i, I) \cdot \mathcal{N}(\mathbf{x}_i(\mathbf{p}) \mid \mathbf{y}_i, \rho\mathcal{I})$, the above optimization problem can be reformulated as:

$$\begin{aligned} \mathbf{p} = \arg\max p(\mathbf{p}) \prod_{i=1}^n \sum_{\mathbf{y}_i \in \mathbf{\Psi}_i} \\ p(l_i = 1 \mid \mathbf{y}_i, I) \mathcal{N}(\mathbf{x}_i(\mathbf{p}) \mid \mathbf{y}_i, \rho\mathcal{I}). \end{aligned} \tag{4}$$

For the case of the prior $p(\mathbf{p})$, which acts as a regularization term, the standard choice is a zero mean Gaussian prior over $\mathbf{q}$ (i.e., $p(\mathbf{p}) = \mathcal{N}(\mathbf{q} \mid \mathbf{0}, \mathbf{\Lambda})$). The above optimization problem was solved in [23] using an Expectation-Maximization (EM) algorithm. The Expectation step concerns the computation of $p(\mathbf{y}_i|l_i = 1, \mathbf{x}_i, I)$, given the parameters $\mathbf{p}$, while the Maximization step involves the minimization of:

$$Q(\mathbf{p}) = ||\mathbf{q}||^{-1}_{\mathbf{\Lambda}} + \sum_{i=1}^{n} \sum_{\mathbf{y}_i \in \mathbf{\Psi}_i} \frac{p(\mathbf{y}_i|l_i = 1, \mathbf{x}_i, I)}{\rho} ||\mathbf{x}_i(\mathbf{p}) - \mathbf{y}_i||^2$$

which can be solved using a Gauss-Newton optimization. This method is known as Regularized Landmark Mean-Shift (RLMS) [23] fitting. Even though it has been shown that the above optimization problem can produce state-of-the-art results it can also suffer from local minimum problem, as all Gauss-Newton optimization methodology.

3. Discriminative Response Map Fitting

In this paper, we follow a different direction to the RLMS approach for the part-based models discussed in the above Section 2.2. Instead of maximizing the probability of a reconstructed shape, given that all parts are correctly located in the image, (i.e., $p(\mathbf{s}(\mathbf{p}) \mid \{l_i = 1\}_{i=1}^n, \mathcal{I})$), we propose to follow a discriminative regression framework for estimating the model parameters $\mathbf{p}$. That is, we propose to find a mapping from the response estimate of shape perturbations to shape parameter updates. In particular, let us assume that in the training set we introduce a perturbation $\Delta\mathbf{p}$ and around each point of the perturbed shape we have response estimates in a $w \times w$ window centered around the perturbed point, $\mathbf{A}_i(\Delta\mathbf{p}) = [p(l_i = 1 \mid \mathbf{x} + \mathbf{x}_i(\Delta\mathbf{p})]$. Then, from the response maps around the perturbed shape $\{\mathbf{A}_i(\Delta\mathbf{p})\}_{i=1}^n$ we want to learn a function f such that $f(\{\mathbf{A}_i(\Delta\mathbf{p})\}_{i=1}^n) = \Delta\mathbf{p}$. We call this the *Discriminative Response Map Fitting (DRMF)* method. The motivation behind this choice was the fact that, contrary to texture features in holistic regression based AAM frameworks [20, 21], response maps (1) can be very well represented by a small set of parameters and (2) learned dictionaries of probability response maps could very faithfully reconstruct response maps in unseen images.

Overall, the training procedure for the DRMF method has two main steps. In the first step, the goal is to train a dictionary for the response map approximation that can be used for extracting the relevant feature for learning the fitting update model. The second step involves iteratively learning the fitting update model which is achieved by a modified boosting procedure. The goal here is to learn a set of weak learners that model the obvious non-linear relationship between the joint low-dimensional projection of the response maps from all landmark points and the iterative 3D shape model parameters update ($\Delta\mathbf{p}$).

3.1. Training Response Patch Model

Before proceeding to the learning step, the goal is to build a dictionary of response maps that can be used for representing any instance of an unseen response map. In other words, our aim is to represent $\mathbf{A}_i(\Delta\mathbf{p})$ using a small number of parameters. Let us assume we have a training set of responses $\{\mathbf{A}_i(\Delta\mathbf{p}_j)\}_{j=1}$ for each point i with various perturbations (including no perturbation, as well). A simple way to learn the dictionary for the i-th point is to vectorize the training set of responses, stack them in a matrix $\mathbf{X}_i = [\text{vec}(\mathbf{A}_i(\Delta\mathbf{p}_1)), \ldots, \text{vec}(\mathbf{A}_i(\Delta\mathbf{p}_n))]$ and since we deal with non-negative responses, the natural choice is to perform Non-negative Matrix Factorization (NMF) [24]. That way the matrix is decomposed into $\mathbf{X}_i \approx \mathbf{Z}_i\mathbf{H}_i$ where $\mathbf{Z}_i$ is the dictionary and $\mathbf{H}_i$ are the weights. Now, given the dictionary $\mathbf{Z}_i$, the set of weights for a response map window $\mathbf{A}_i$ for the point i can be found by:

$$\mathbf{h}_i^o = \arg\max_{\mathbf{h}_i} ||\mathbf{Z}_i\mathbf{h}_i - \text{vec}(\mathbf{A}_i)||^2, \text{ s.t } \mathbf{h}_i \geq 0 \quad (5)$$

which can be solved using NMF strategies [24]. Then, instead of finding a regression function from the perturbed responses $\{\mathbf{A}_i(\Delta\mathbf{p})\}_{i=1}^n$, we aim at finding a function from the low-dimensional weight vectors $\{\mathbf{h}_i(\Delta\mathbf{p})\}_{i=1}^n$ to the update of parameters $\Delta\mathbf{p}$.

For practical reasons and to avoid solving the optimization problem (5) for each part in the fitting procedure, instead of NMF we have also applied PCA on $\{\mathbf{A}_i(\Delta\mathbf{p}_j)\}_{j=1}^N$. Using PCA, the extraction of the corresponding weigh vector $\mathbf{h}_i$ can be performed very efficiently by just a simple projections on the PCA bases. An illustrative example on how effectively a response map can be reconstructed by as small number of PCA components (capturing 85% of the variation) is shown in Figure 1. We refer to this dictionary as *Response Patch Model* represented by:

$$\{\mathcal{M}, \mathcal{V}\} : \mathcal{M} = \{\mathbf{m}_i\}_{i=1}^n \text{ and } \mathcal{V} = \{\mathbf{V}_i\}_{i=1}^n \quad (6)$$

where, $\mathbf{m}_i$ and $\mathbf{V}_i$ are the mean vector and PCA bases, respectively, obtained for each of the n landmark points.

3.2. Training Parameter Update Model

Given a set of N training images $\mathcal{I}$ and the corresponding shapes $\mathcal{S}$, the goal is to iteratively model the relationship between the joint low-dimensional projection of the response patches, obtained from the response patch model $\{\mathcal{M}\ ,\ \mathcal{V}\}$, and the parameters update ($\Delta\mathbf{p}$). For this, we propose to use a modified boosting procedure in that we uniformly sample the 3D shape model parameter space within a pre-defined range around the ground truth parameters $\mathbf{p}_g$ (See Eqn. 1), and iteratively model the relationship between the joint low-dimensional projection of the response patches at the current sampled shape (represented by t^{th} sampled shape parameter $\mathbf{p}_t$) and the parameter update $\Delta\mathbf{p}$ ($\Delta\mathbf{p} = \mathbf{p}_g - \mathbf{p}_t$). The step-by-step training procedure is as follow:

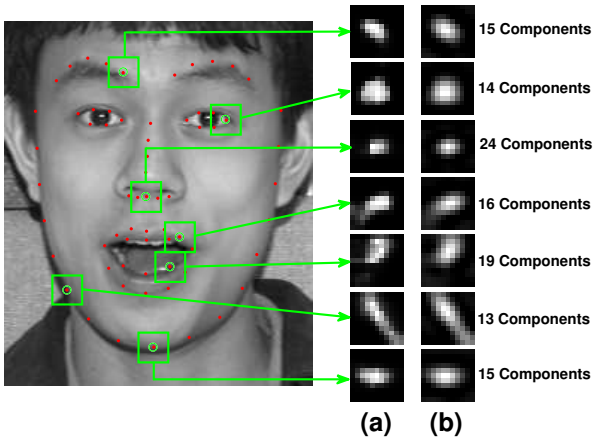

Figure 1. Overview of the response patch model: (a) Original HOG based response patches. (b) Reconstructed response patches using the response patch model that captured 85% variation.

Let T be the number of shape parameters set sampled from the shapes in $\mathcal{S}$, such that the initial sampled shape parameter set is represented by $\mathcal{P}^{(1)}$:

$$\mathcal{P}^{(1)} = \{\mathbf{p}_j^{(1)}\}_{j=1}^{T} \quad \text{and} \quad \psi^{(1)} = \{\Delta\mathbf{p}_j^{(1)}\}_{j=1}^{T} \tag{7}$$

'1' in the superscript represents the initial set (first iteration). Next, extract the *response patches* for the shape represented by each of the sampled shape parameters in $\mathcal{P}^{(1)}$ and compute the low-dimensional projection using the response patch model $\{\mathcal{M}, \mathcal{V}\}$. Then, concatenate the projections to generate a joint low-dimensional projection vector $\mathbf{c}(\Delta\mathbf{p}_j^{(1)}) = [\mathbf{h}_1(\Delta\mathbf{p}_j^{(1)}), \ldots, \mathbf{h}_n(\Delta\mathbf{p}_j^{(1)})]^T$, one per sampled shape, such that:

$$\chi^{(1)} = \{\mathbf{c}(\Delta\mathbf{p}_j^{(1)})\}_{j=1}^{T} \tag{8}$$

where, $\chi^{(1)}$ represents the initial set of joint low-dimensional projections obtained from the training set. Now, with the training set $\mathcal{T}^{(1)} = \{\chi^{(1)}, \psi^{(1)}\}$, we learn the fitting parameter update function for the first iteration i.e. a weak learner $\mathcal{F}^{(1)}$:

$$\mathcal{F}^{(1)} : \psi^{(1)} \leftarrow \chi^{(1)} \tag{9}$$

We then propagate all the samples from $\mathcal{T}^{(1)}$ through $\mathcal{F}^{(1)}$ to generate $\mathcal{T}_{new}^{1}$ and eliminate the converged samples in $\mathcal{T}_{new}^{(1)}$ to generate $\mathcal{T}^{(2)}$ for the second iteration. Here, convergence means that the shape root mean square error (RMSE) between the predicted shape and the ground truth shape is less than a threshold (for example, set to 2 for the experiments in this paper). Any regression method can be employed in our framework. We have chosen a simple Linear Support Vector Regression (SVR) [15] for each of the shape parameters. In total, we used 16 shape parameters i.e. 6 global shape parameters and the top 10 non-rigid shape parameters. Structured regression based approaches can also be employed but we opted to show the power of our method with a very simple regression frameworks.

In order to replace the *eliminated* converged samples, we generate a new set of samples (Eqn. 7 and Eqn. 8) from the same images in $\mathcal{I}$ whose samples converged in the first iteration. We propagate this new sample set through $\mathcal{F}^1$ and eliminate the converged samples to generate an additional *replacement* training set for the second iteration $\mathcal{T}_{rep}^{(2)}$. The training set for the second iteration is updated:

$$\mathcal{T}^{(2)} \leftarrow \{\mathcal{T}^{(2)}, \mathcal{T}_{rep}^{(2)}\} \tag{10}$$

and the fitting parameter update function for the second iteration is learnt i.e. a weak learner $\mathcal{F}^{(2)}$. The sample elimination and replacement procedure for every iteration have two-fold benefits. Firstly, it plays an important role in insuring that the progressive fitting parameter update functions are trained on the tougher samples that have not converged in the previous iterations. And secondly, it helps in regularizing the learning procedure by correcting the samples that diverged in the previous iterations due to overfitting.

The above training procedure is repeated iteratively until all the training samples have converged or the maximum number of desired training iterations (η) have been reached. The resulting fitting parameter update model $\mathcal{U}$ is a set of weak learners:

$$\mathcal{U} = \{\mathcal{F}^{(1)}, \ldots, \mathcal{F}^{(\eta)}\} \tag{11}$$

The training procedure is outlined in Algorithm 1.

Algorithm 1: Training Parameter Update Model

Require: PDM (Eqn. 1), $\mathcal{I}$, $\mathcal{S}$, $\{\mathcal{M}, \mathcal{V}\}$ (Eqn. 6).

1. Get initial shape parameters sample set (Eqn. 7).
2. Get initial joint low-dimensional projection set (Eqn. 8).
3. Generate training set for first iteration $\mathcal{T}^{(1)}$.
4. **for** $i = 1 \rightarrow \eta$ **do**
5. Compute the weak learner $\mathcal{F}^{(i)}$ using $\mathcal{T}^{(i)}$.
6. Propagate $\mathcal{T}^{(i)}$ through $\mathcal{F}^{(i)}$ to generate $\mathcal{T}_{new}^{(i)}$.
7. Eliminate converged samples in $\mathcal{T}_{new}^{(i)}$ to generate $\mathcal{T}^{(i+1)}$.
8. **if** $\mathcal{T}^{(i+1)}$ *is empty* **then**
9. All training samples converged. *Stop Training*.
10. **else**
11. Get new shape parameters sample set (Eqn. 7) from images whose samples are eliminated in Step 7.
12. Get new joint low-dimensional projection set (Eqn. 8) for the samples generated in Step 11.
13. Generate new *replacement* training set $\mathcal{T}_{rep}^{(i)}$.
14. **for** $j = 1 \rightarrow (i-1)$ **do**
15. Propagate $\mathcal{T}_{rep}^{(i)}$ through $\mathcal{F}^{(j)}$.
16. Eliminate converged samples in $\mathcal{T}_{rep}^{(i)}$.
17. Update $\mathcal{T}^{(i+1)} \leftarrow \{\mathcal{T}^{(i+1)}, \mathcal{T}_{rep}^{(i)}\}$

Output : Fitting Parameter Update Model $\mathcal{U}$ (Eqn. 11).

3.3. Fitting Procedure

Given the test image $\mathcal{I}_{test}$, the fitting parameter update model $\mathcal{U}$ is used to compute the additive parameter update $\Delta\mathbf{p}$ iteratively. The *goodness of fitting* is judged by the fitting score that is computed for each iteration by simply adding the responses (i.e. the probability values) at the landmark locations estimated by the current shape estimate of that iteration. The final fitting shape is the shape with the highest fitting score.

4. Experiments

We conducted generic face fitting experiments on the Multi-PIE [14], XM2VTS [19] and the LFPW [6] databases. The Multi-PIE database is the most commonly used database for generic face fitting and is the best for comparison with previous approaches. Moreover, its consists of thousands of images with combined variations of identity, expression, illumination and pose, making it a very useful database for highlighting the ability of the proposed DRMF method (Section 3) to handle all these combined variations accurately in the generic face fitting scenario. The XM2VTS database focuses mainly on the variations in identity and is a challenging database in a generic face fitting scenario because of the large variations in facial shape and appearance due to facial hair, glasses, ethnicity and other subtle variations. Unlike the Multi-PIE and the XM2VTS, the LFPW database is a completely *wild* database, i.e. consists of images captured under uncontrolled natural settings, and is an extremely challenging database for the generic face fitting experiment.

For all the experiments, we consider the independent model (p1050) of the tree-based method [26], released by the authors, as the baseline method for comparison. For the multi-view CLM approach, the pose range of $\pm 30^\circ$ in yaw (i.e. with pose code 051, 050, 140, 041 and 130) is divided into three view-based CLMs with each covering -30° to -15°, -15° to 15° and 15° to 30° in yaw, respectively. Other non-frontal poses have been excluded from our experiment for the lack of ground-truth annotations.

Another consistent aspect for all the following experiments is the initialization of the fitting procedure. For CLMs, we directly used the off-the-shelf OpenCV face detector. However, this face detector often fails on the LFPW dataset and for several images with varying illumination and pose in Multi-PIE and XM2VTS database. Therefore, for the images on which the face detector failed, we used the bounding box provided by our own trained tree-based model p204 (described in the following section) and perturbed this bounding box by 10 pixels for translation, 5° for rotation and 0.1 for scaling factor. We then initialized the mean face at the centre of this perturbed bounding box.

Overview of Results:

[1] The Multi-PIE experiment focuses on accessing the performance with combined identity, pose, expression and illumination variation. The results show significant performance gain for the proposed DRMF method over all other methods. Furthermore, the results show that the CLMs outperform the equivalent tree-based model for the task of landmark localization. We believe this is due to the use of tree-based shape model that allows for non-face like structures to occur making it hard to accurately fit the model, especially for the case of facial expressions.

[2] XM2VTS experiment, performed in an out-of-database scenario, highlights the ability of the DRMF method to handle unseen variations and other challenging variations like facial hair, glasses and ethnicity.

[3] LFPW experiment further verifies the generalization capability of the DRMF method to handle challenging uncontrolled natural variations. The results show that DRMF outperform RLMS and the tree-based method [26] convincingly on this wild database.

[4] The results on XM2VTS and LFPW database also validate one of the main motivations behind the DRMF method i.e. the response maps extracted from an unseen image can be very faithfully represented by a small set of parameters and are suited for the discriminative fitting frameworks, unlike the holistic texture based features.

[5] Moreover, the fitting procedure of the DRMF method is highly efficient and is real-time capable. The current MATLAB implementation of the Multiview DRMF method, using the HOG feature based patch experts, takes 1 second per image on Intel Xeon 3.80 GHz processor. We release the source code[1] and the pre-trained models for the research purposes.

4.1. Multi-PIE Experiments

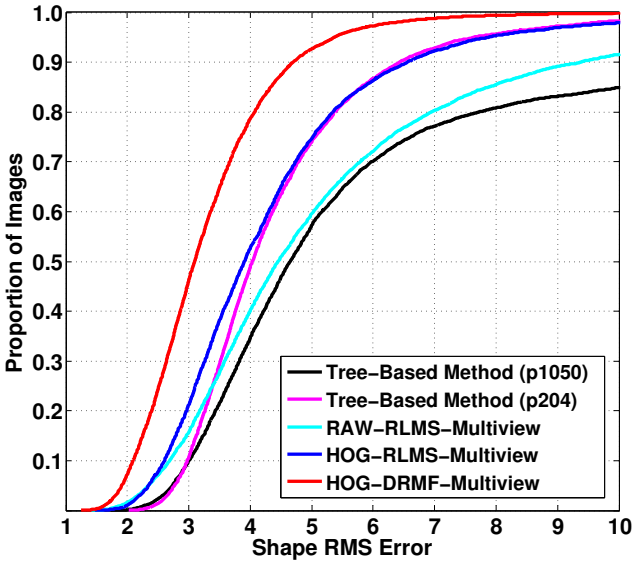

Figure 2. Experiment on Multi-PIE database.

The goal of the this experiment is to compare the performance of the HOG feature based CLM framework, using the RLMS [23] and the proposed DRMF (Section 3) method, with the tree-based method [26] under combined variations of identity, pose, expression and illumination. For this, images of all 346 subjects with all six expressions at frontal and non-frontal poses at various illumination conditions are used. The training set consisted of roughly 8300 images which included the subjects 001-170 at poses 051, 050, 140, 041 and 130 with all six expressions at frontal illumination and one other randomly selected illumination condition. For this experiment, we train several versions of the CLMs described below. The multi-view CLMs trained using the HOG feature based patch experts and the RLMS fitting method is referred as HOG-RLMS-Multiview. Whereas, the multi-view CLMs trained using the HOG feature based patch experts and the DRMF fitting method (Section 3) is referred as as HOG-DRMF-Multiview. Moreover, we also trained RAW-RLMS-Multiview which refers to the multi-view CLM using the RAW pixel based patch experts and the RLMS fitting method. This helps in showing the performance gained by using the HOG feature based patch experts instead of the RAW pixel based patch experts.

For the tree-based method [26], we trained the tree-based model p204 that share the patch templates across the neighboring viewpoints and is equivalent to the multi-view CLM methods, using exactly the same training data for a fair comparison with CLM based approaches. We did not train the independent tree-based model (equivalent to p1050) because of its unreasonable training requirements, computational complexity and limited practical utility. Basically, training an independent tree-based model amounts to training separate models for each variation present in the dataset i.e. different models for every pose and expression. For our dataset that consists of five poses with all six expressions, an independent tree-based model will require training 2050 part detectors (i.e. 68 points $\times$ 5 poses $\times$ 6 expressions = 2050 independent parts). With preliminary calculations, such a model will require over a month of training time and nearly 90 seconds per image of fitting time.

The test set consisted of roughly 7100 images which included the subjects 171-346 at poses 051, 050, 140, 041 and 130 with all six expressions at frontal illumination and one other randomly selected illumination condition. From the results in Figure 2, we can clearly see that the HOG-DRMF-Multiview outperforms all other method by a substantial margin. We also see a substantial gain in the performance by using the HOG feature based patch experts (HOG-RLMS-Multiview) instead of the RAW pixel (RAW-RLMS-Multiview). Moreover, the HOG-RLMS-Multiview also outperform the equivalent tree-based model p204 for the task of landmark localization. The qualitative analysis of the results suggest that the tree-based methods [26], although suited for the task of face detection and rough pose estimation, are not well suited for the task of landmark localization. We believe, this is due to the use of tree-based shape model that allows for the non-face like structures to occur frequently, especially for the case of facial expressions. See the sample fitting results in Figure 5.

4.2. XM2VTS Experiments

All 2360 images from XM2VTS database [19] were manually annotated with the 68-point markup and are used as the test set. This experiment is performed in an out-of-database scenario i.e. the models used for fitting are trained entirely on the Multi-PIE database. We used the HOG-DRMF-Multiview, HOG-RLMS-Multiview and the tree-based model p204, used for generating results in Figure 2, to perform the fitting on the XM2VTS database. Note that this database consists of only frontal images. Nonetheless, the results from Figure 3 show that the HOG-DRMF-Multiview outperforms all other methods again. Moreover, the HOG-RLMS-Multiview outperforms the tree-based model p204 and the baseline p1050 convincingly.

This results is particularly important because it highlights the capability of the DRMF method to handle unseen variations. The generative model based discriminative approaches [16, 20, 21] have been reported to generalize well for the variations present on the training set, however, the overall performance of these discriminative fitting methods have been shown to deteriorate significantly for out-of-database experiments [22]. The results show that not only does DRMF outperform other state-of-the-art approaches in an out-of-database experiment but also handles the challenging variations in the facial shape and appearance present in the XM2VTS database due to facial hair, glasses and ethnicity. This result validates one of the main motivations behind the DRMF method i.e. the response maps extracted from an unseen image can be very faithfully represented by a small set of parameters and are suited for the discriminative fitting frameworks, unlike the holistic texture based features.

4.3. LFPW Experiments

For further test the ability of the DRMF method to handle unseen variations, we conduct experiments using the database that presents the challenge of uncontrolled natural settings. The Labeled Face Parts in the Wild (LFPW) database [6] consist of the URLs to 1100 training and 300 test images that can be downloaded from internet. All of these images were captured *in the wild* and contain large variations in pose, illumination, expression and occlusion. We were able to download only 813 training images and 224 test images because some of the URLs are no longer valid. These images were manually annotated with the 68-point markup to generate the ground-truths used in this section.

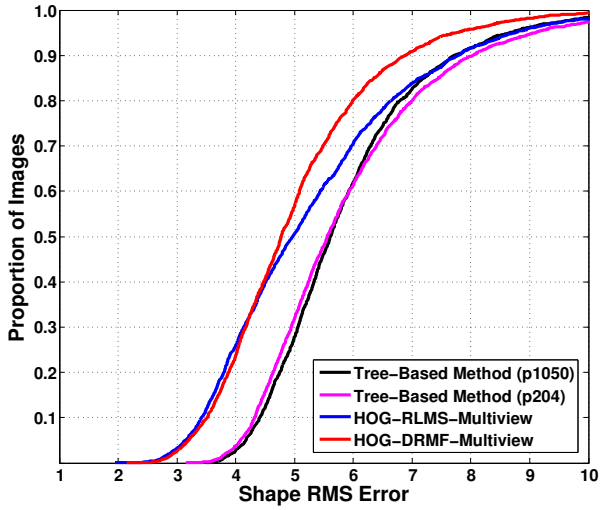

Figure 3. Out-of-database experiment on XM2VTS database.

We used the HOG-DRMF-Multiview, HOG-RLMS-Multiview and the tree-based model p204 trained only on the Multi-PIE database (used previously for generating results in Figure 2) to perform fitting on the LFPW test set. We then augmented the Multi-PIE training set with the LFPW training set and re-trained the CLM and tree-based models. We refer to these methods as HOG-Wild-DRMF-Multiview, HOG-Wild-RLMS-Multiview and the tree-based model p204-Wild. These wild models were then used to perform fitting on the LFPW test set and the results are reported in Figure 4. Note that the size of the faces in these images vary greatly because of the wild nature of this dataset. Therefore, we normalized the shape RMSE by the distance between the eye-corners which we believe is the best way to show unbiased results. From these results, we can clearly see the dominance of the HOG-Wild-DRMF-Multiview over other methods.

Firstly, this result clearly show that the proposed response map based discriminative fitting methodology can handle *wild face* and further emphasises the suitability of the parameterized response map models for the discriminative fitting frameworks. Secondly, an interesting result is the performance gain achieved by augmenting the Multi-PIE training set with the LFPW training set. Notice that in Figure 4, the accuracy of HOG-Wild-DRMF-Multiview increases consistently in comparison to the HOG-DRMF-Multiview (for example, by over 13% for the cases with Shape RMSE below 0.05 fraction of inter-ocular distance). Whereas for the same scenario, HOG-Wild-RLMS-Multiview show little improvement in performance over HOG-RLMS-Multiview (for example, increases by a little over 2% for the cases with Shape RMSE below 0.05 fraction of inter-ocular distance). This shows the advantage of the proposed response map based discriminative fitting approach that uses the available training data in a more useful way by learning the fitting update model as compared to the RLMS that rely entirely on the gauss-newton optimization based methodologies.

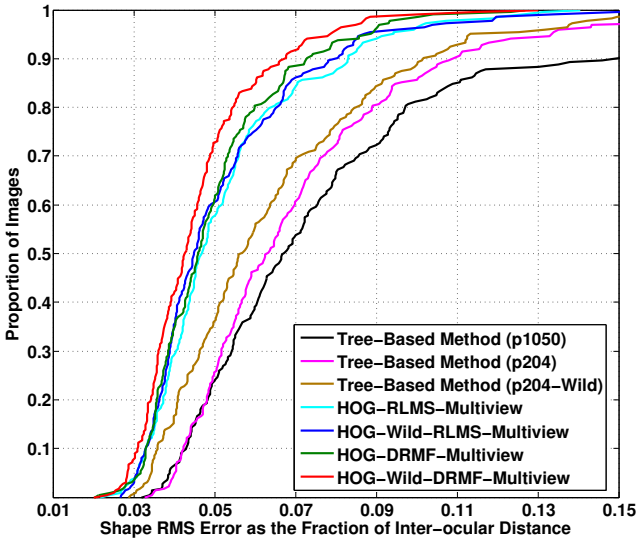

Figure 4. Wild experiments on LFPW database.

5. Conclusion

We have presented a novel Discriminative Response Map Fitting (DRMF) method for the CLM framework. We conduct detailed experiments in a generic face fitting scenario on the databases with images captured under both the controlled (Multi-PIE and XM2VTS) and uncontrolled natural setting (LFPW Database). The results show that the proposed DRMF method outperforms the state-of-the-art RLMS fitting method [23] and the recently proposed tree-based method [26] consistently across all databases. See the sample fitting results in Figure 5. Moreover, the DRMF method is computationally very efficient and real-time capable. The current MATLAB implementation takes 1 second per image on an Intel Xeon 3.80 GHz processor. We release the MATLAB code[1] for the multi-view HOG-CLM framework with the DRMF method and the pre-trained models for research purposes.

Acknowledgments : This work has been funded by the European Research Council under the ERC Starting Grant agreement no. ERC-2007-StG-203143 (MAHNOB). The work of Akshay Asthana is funded by Marie Curie International Incoming Fellowship under the FP7-PEOPLE-2011-IIF Grant agreement no. 302836 (FER in the Wild).

References

[1] T. Albrecht, M. Lüthi, and T. Vetter. A statistical deformation prior for non-rigid image and shape registration. In *CVPR*, 2008. 2

(a) DRMF Fitting Results

(b) RLMS Fitting Results

(c) Tree-based Model (p204) Fitting Results

Figure 5. Sample Fitting Results. Column 1-3: Multi-PIE Results. Column 4-7: LFPW Results.

[2] B. Amberg, A. Blake, and T. Vetter. On compositional image alignment, with an application to active appearance models. In *IEEE CVPR*, 2009. 2

[3] A. Asthana, J. Saragih, M. Wagner, and R. Goecke. Evaluating AAM fitting methods for facial expression recognition. In *ACII*, 2009. 1

[4] S. Baker, R. Gross, and I. Matthews. Lucas-Kanade 20 years on: A unifying framework: Part 3. Technical report, RI, CMU, USA, 2003. 1, 2

[5] S. Baker and I. Matthews. Equivalence and Efficiency of Image Alignment Algorithms. In *CVPR*, 2001. 1, 2

[6] P. N. Belhumeur, D. W. Jacobs, D. J. Kriegman, and N. Kumar. Localizing parts of faces using a consensus of exemplars. In *CVPR*, 2011. 1, 5, 6

[7] V. Blanz and T. Vetter. Face Recognition Based on Fitting a 3D Morphable Model. *IEEE PAMI*, 25(9):1063–1074, Sept. 2003. 1, 2

[8] X. Cao, Y. Wei, F. Wen, and J. Sun. Face alignment by explicit shape regression. In *CVPR*, 2012. 2

[9] T. Cootes, G. Edwards, and C. Taylor. Active Appearance Models. In *ECCV*, 1998. 1, 2

[10] T. Cootes, C. Taylor, D. Cooper, and J. Graham. Active shape models - their training and applications. *CVIU*, 1995. 1, 2

[11] D. Cristinacce and T. Cootes. Feature detection and tracking with constrained local models. In *BMVC*, 2006. 1

[12] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *CVPR*, 2005. 1

[13] G. Edwards, C. Taylor, and T. Cootes. Interpreting Face Images Using Active Appearance Models. In *IEEE FG*, 1998. 1

[14] R. Gross, I. Matthews, J. Cohn, T. Kanade, and S. Baker. Multi-PIE. In *IEEE FG*, 2008. 1, 5

[15] C. Ho and C. Lin. Large-scale linear support vector regression. Technical report, Technical report, NTU, 2012. 4

[16] X. Liu. Discriminative face alignment. *IEEE PAMI*, 31(11):1941–1954, Nov. 2009. 1, 2, 6

[17] I. Matthews and S. Baker. Active Appearance Models Revisited. *IJCV*, 60(2):135–164, Nov. 2004. 1, 2

[18] I. Matthews, J. Xiao, and S. Baker. 2D vs. 3D Deformable Face Models: Representational Power, Construction, and Real-Time Fitting. *IJCV*, 75(1):93–113, Oct. 2007. 2

[19] K. Messer, J. Matas, J. Kittler, J. Lttin, and G. Maitre. Xm2vtsdb: The extended m2vts database. In *AVBPA*, 1999. 1, 5, 6

[20] J. Saragih and R. Goecke. Iterative Error Bound Minimisation for AAM Alignment. In *ICPR*, 2006. 1, 2, 3, 6

[21] J. Saragih and R. Goecke. A Nonlinear Discriminative Approach to AAM Fitting. In *ICCV*, 2007. 1, 2, 3, 6

[22] J. Saragih and R. Goecke. Learning AAM fitting through simulation. *Pattern Recognition*, 42(11):2628–2636, Nov. 2009. 1, 6

[23] J. Saragih, S. Lucey, and J. Cohn. Deformable model fitting by regularized landmark mean-shift. *IJCV*, 91(2):200–215, Jan. 2011. 1, 2, 3, 6, 7

[24] D. Seung and L. Lee. Algorithms for non-negative matrix factorization. *Advances in neural information processing systems*, 13:556–562, 2001. 3

[25] G. Tzimiropoulos, J. Alabort-i-medina, S. Zafeiriou, and M. Pantic. Generic active appearance models revisited. In *ACCV*, 2012. 2

[26] X. Zhu and D. Ramanan. Face detection, pose estimation and landmark localization in the wild. In *CVPR*, 2012. 1, 2, 5, 6, 7