Mnemonic Descent Method: A recurrent process applied for end-to-end face alignment

George Trigeorgis[⋆] Patrick Snape[⋆] Mihalis A. Nicolaou[†] Epameinondas Antonakos[⋆]
Stefanos Zafeiriou[⋆,∗]
[⋆]Department of Computing, Imperial College London, UK
[†]Department of Computing, Goldsmiths, University of London, UK
[∗]Center for Machine Vision and Signal Analysis, University of Oulu, Finland
[⋆]{g.trigeorgis, p.snape, e.antonakos, s.zafeiriou}@imperial.ac.uk, [†]m.nicolaou@gold.ac.uk

Abstract

Cascaded regression has recently become the method of choice for solving non-linear least squares problems such as deformable image alignment. Given a sizeable training set, cascaded regression learns a set of generic rules that are sequentially applied to minimise the least squares problem. Despite the success of cascaded regression for problems such as face alignment and head pose estimation, there are several shortcomings arising in the strategies proposed thus far. Specifically, (a) the regressors are learnt independently, (b) the descent directions may cancel one another out and (c) handcrafted features (e.g., HoGs, SIFT etc.) are mainly used to drive the cascade, which may be sub-optimal for the task at hand. In this paper, we propose a combined and jointly trained convolutional recurrent neural network architecture that allows the training of an end-to-end to system that attempts to alleviate the aforementioned drawbacks. The recurrent module facilitates the joint optimisation of the regressors by assuming the cascades form a non-linear dynamical system, in effect fully utilising the information between all cascade levels by introducing a memory unit that shares information across all levels. The convolutional module allows the network to extract features that are specialised for the task at hand and are experimentally shown to outperform hand-crafted features. We show that the application of the proposed architecture for the problem of face alignment results in a strong improvement over the current state-of-the-art.

1. Introduction

Non-linear least squares optimisation problems often arise in computer vision, including but not limited to Structure-from-Motion [9, 23], rigid and deformable image alignment [33, 35], optical flow estimation [33, 66, 46, 45], and estimation of camera parameters for calibration [44]. The application of standard Newton-type methods for retrieving the optimal parameters is challenging due to the highly non-convex nature of the cost functions and the lack of differentiability of commonly used image operators (such as HoGs [17], SIFT [32], etc.). Recently, in order to address the drawbacks of Newton/Gauss-Newton methods a set of generic "descent directions" is learnt through the application of a cascade of regressors [56, 11]. Generally, these directions are learnt independently per cascade via simulation. That is, in the case of deformable face alignment, the ground-truth facial shapes of the training images are randomly perturbed (according to a fixed variance). The descent directions are then estimated independently and seek to progress from the perturbed shapes to the ground-truth. In their simplest form, these rules can be learnt through the application of successive stages of linear regression, each minimising the average error over all training samples. The use of regression/learning based-methods for solving non-linear optimisation problems has a

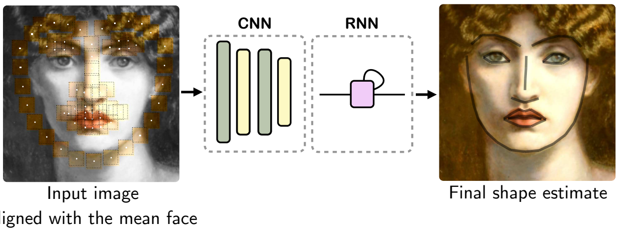

Figure 1: Mnemonic Descent Method learns to align a shape estimate to a facial image[1] in an end-to-end manner using a jointly learnt convolutional recurrent network architecture.

[1] Depicted is the muse Mnemosene which was the personification of memory in Greek mythology. The painting is an interpretation of the muse by the the father of the Pre-Raphaelite brotherhood Dante Gabriel Rossetti.

rich history in computer vision, beginning with the first Active Appearance Models (AAMs) [13], where average Jacobians were learnt from the training set[2]. Cascaded regression methodologies have been also proposed in the recent works of [20, 11, 4, 49, 37, 26, 50]. However, to the best of our knowledge, the only work that proposes a generic framework for solving non-linear least squares is the so-called Supervised Descent Method (SDM) [56, 57].

Several shortcomings can be identified in the state-of-the-art cascade methods for deformable face alignment:

- The cascade steps are learnt independently. Each linear regressor simply learns how to regress from a particular fixed variance of shape perturbations to the ground-truth [3]. Thus, correlations between semantically related image characteristics, such as facial pose, are not taken into account.
- The result of the optimisation is tightly coupled with the image features chosen to drive the regression. Hand-crafted features are not data-driven and thus not necessarily optimal for the face alignment task. In contrast, binary/tree-based features [37, 26, 11, 20] are data-driven and have shown to be effective for face alignment. However, these simple pixel intensity differences can not be learnt in an end-to-end manner. The success seen by convolutional features for various computer vision tasks has yet to be realised for face alignment. In particular, no currently proposed system trains end-to-end convolutional features.

In this paper, we propose the Mnemonic Descent Method (MDM) to address the issues above. In particular, MDM models deformable face alignment as a non-linear dynamical system[3]. MDM maintains an internal memory unit that accumulates information extracted from the history of *all* past observations of the input space. This has the advantage that descent directions are naturally partitioned according to the previously calculated descent directions. This paradigm maps to a very intuitive justification when applied to the problem of face alignment. For example, it seems reasonable that the alignment of any near profile face from a frontal initialisation will have an extremely similar sequence of descent directions. This is validated experimentally in Fig. 3. MDM leverages this rich information and trains an end-to-end face alignment method that learns a set of data driven features, using a Convolutional Neural Network (CNN), directly from the images in a cascaded manner and most importantly uses a Recurrent Neural Network (RNN) to impose a memory constraint on the descent directions as illustrated in Fig.1. Our work is also motivated by the success of end-to-end training of convolutional recurrent networks for the tasks of image caption generation [12], scene parsing [36], and image retrieval/annotation generation [21]. To the best of our knowledge this is the first end-to-end recurrent convolutional system for deformable object alignment. In summary, the contributions of this work are:

1. We propose a non-linear cascaded framework for end-to-end learning of the descent directions of non-linear functions. These types of functions are widely applicable in computer vision and existing works such as SDM [56] have shown that descent direction learning can be highly effective.
2. This is the first work on face alignment where a single model is trained end-to-end i.e. from the raw image pixels to the final predictions. We incorporate problem-specific information in the training procedure by learning new image features via a CNN.
3. We introduce the concept of memory into descent direction learning. We believe this is highly discriminative and one of the major strengths of our approach.
4. We improve on the state-of-the-art in face alignment on the challenging test-set of 300W competition [38, 39] by a large margin.

The remainder of this paper is organised as follows. In Sec.2, we provide an overview of related work, with particular emphasis on SDM (Sec.3). Subsequently, in Sec.4, we introduce the proposed Mnemonic Descent Method (MDM) and, without loss of generality, describe its application to face alignment. Finally, in Sec.5, we provide rigorous evaluations of our model, in order to demonstrate the advantages of the proposed MDM over the state-of-the-art.

2. Related Work

The area of deformable face alignment constitutes a very intuitive domain for the application of this work and is thus chosen as the main application domain for evaluation.

Face alignment has a long and rich history that includes the introduction of many important works in computer vision such as Active Appearance Models [13, 35, 2], Constrained Local Models [16, 41] and 3D Morphable Models [8]. In recent years, the problem of face alignment has seen substantial improvement, partially due to the introduction of large datasets of unconstrained (in-the-wild) images [6, 29, 65], which have been consistently

[2] The team led by Prof. Tim Cootes has proposed many variants for learning descent directions [13, 49, 14, 15]

[3] The only alignment method that uses a dynamical system, and in particular a Linear Dynamical System (LDS), to model the shape estimates during model fitting is [34]. The LDS is used to infer the posterior distribution of the global warp and used in a Constrained Local Model (CLM) framework. CLMs have not achieved state-of-the-art results in recent challenges such as the 300-W competition [38, 39], even when trained in a cascaded regression [4].

re-annotated [39, 40, 38]. This increase in data variability and quantity has expanded the power of discriminative methods, such as regression based methods. In particular, many recently successful techniques chain a number of regressors together sequentially, in what is commonly called a *cascade*. Cascaded regression strategies constitute a large portion of the most popular facial alignment algorithms, as they are highly efficient and generalise well [56, 37, 11, 10, 3, 26, 31, 64, 57, 50]. The most efficient cascaded regression methods are those that achieve regression via boosting of weak learners such as random ferns [11, 10] or random forests [37, 26]. However, a seminal work in this area which generalises to a multitude of problems and can efficiently deal with a large battery of non-linear least squares problems, is that of SDM [56, 57]. SDM was the first work to describe the cascaded regression problem as a more general learning framework in terms of optimising non-linear objective functions utilising learnt descent directions from training data. In particular, the regressors at each cascade are assumed to be linear and model average descent directions in the space of the objective function. However, the learnt descent directions, despite being chained in a cascade, are only related to one another via the variance remaining from the previous cascade. Therefore, the initial cascade levels are prone to large descent steps which may not generalise well. This was addressed in [57] by partitioning the descent directions into cohesive groups during training. At test time, a partition is chosen that represents the correct descent direction. For example, for face alignment this requires an initial estimate of the shape and the descent directions are partitioned according to facial pose. However, this implies that [57] is only useful for tracking scenarios where the previous frame provides the prior information for selecting the correct partition.

Asthana *et al.* [3] proposed an incremental learning framework for SDM type methods which supports the total independence of each cascade level. They assume that each cascade is independent and therefore cascade levels can be learnt in parallel by merely *simulating the residual variance remaining after applying the previous cascade*. Although the independence of each level may be attractive for incremental learning, we propose that descent directions should be influenced by prior knowledge from previous descent steps. We propose to model the procedure as a non-linear dynamical system where a continuous latent state variable appropriately drives the procedure. In this paper, we show that it is possible to obtain large improvements when, instead of utilising hand-crafted features, optimal features for the given problem are learnt in an end-to-end fashion.

Our proposed method is also reminiscent of previously proposed deep learning methods for face alignment [43, 55, 47, 61, 63, 62]. Sun *et al.* [47] and Zhou *et al.* [63] propose to use independent Convolutional Neural Networks (CNN) to perform coarse-to-fine shape searching. Zhang *et al.* [61] also utilise a coarse-to-fine shape search using first a global and then a set of local stacked autoencoders. However, each autoencoder is trained in isolation. Zhang *et al.* [62] propose a novel approach that involves incorporating auxiliary information into the fitting process. Unlike other related methods, they do not incorporate a cascade of networks but instead frame the problem as a multi-task learning problem. Wu *et al.* [55] use a deep belief network to train a more flexible shape model, but do not learn any convolutional features. Finally, Baoguang Shi *et al.* [43] propose to jointly learn a cascade of linear regressors. Although the regressors are updated jointly via back-propagation, [43] uses linear regressors and employs hand crafted HoG features [17] rather than learning the features directly from the images. Also, via close inspection of the results reported in [43], we found that their joint cascade methodology did not lead to any improvements in alignment accuracy over cascade regression methods that were trained independently, e.g. 6.32 mean error on the 300W fullset [39, 38] for [37] vs. 6.31 for [43]. In the following section (Sec. 3), we formally introduce the face alignment problem and provide a brief description of the SDM algorithm.

3. Cascaded Regression

Face alignment is defined as the problem of localising a set of l sparse fiducial points, $\mathbf{l}_i = [x_i, y_i]^\top$ on an image, $I \in \mathbb{R}^{w \times h}$, of a face. Given an image and an initial estimate of the shape within the image, $\mathbf{x}^{(0)} = [\mathbf{l}_1{}^\top, \ldots, \mathbf{l}_l{}^\top]^\top$ where $\mathbf{x}^{(0)} \in \mathbb{R}^{d \times 1}$ with $d = 2l$, face alignment seeks to recover the ground-truth facial shape $\mathbf{x}^*$. In the case of cascaded regression methods such as SDM, the optimisation from $\mathbf{x}^{(0)}$ to $\mathbf{x}^*$ is learnt from a large training set of images by successively learning a series of linear regressors. Most commonly, the regression parameters are optimised based on a set of complex features extracted from each image around the local area of each of the l fiducial points. We denote the extraction of these features for fixed sized patches (local square regions) from an image as $\phi(I_i; \mathbf{x}_i) \in \mathbb{R}^{f \times 1}$. Since SDM proposes to learn a cascade of regressors, the target variables for regression are expressed as shape increments, defined by $\Delta\mathbf{x}_i^{(k)} = \mathbf{x}_i^* - \mathbf{x}_i^{(k)}$, where k is the current cascade index and, thus, $\mathbf{x}_i^{(k)}$ is the current shape estimate of the i-th image. Finally, given n input training images, the design matrix is formulated as $\mathbf{\Phi} = [\phi(I_1; \mathbf{x}_1), \ldots, \phi(I_n; \mathbf{x}_n)]$ where $\mathbf{\Phi} \in \mathbb{R}^{f \times n}$. The matrix encapsulating the target shape increments is also denoted as $\Delta\mathbf{X} = [\Delta\mathbf{x}_1, \ldots, \Delta\mathbf{x}_n]$ where $\Delta\mathbf{X} \in \mathbb{R}^{d \times n}$. SDM [56] proposes to learn a series of k linear regressions formulated as

$$\underset{\mathbf{R}^{(k)}}{\arg\min} \|\Delta\mathbf{X}^{(k)} - \mathbf{R}^{(k)}\left[\mathbf{\Phi}^{(k)}\mathbf{1}\right]\|_F^2, \tag{1}$$

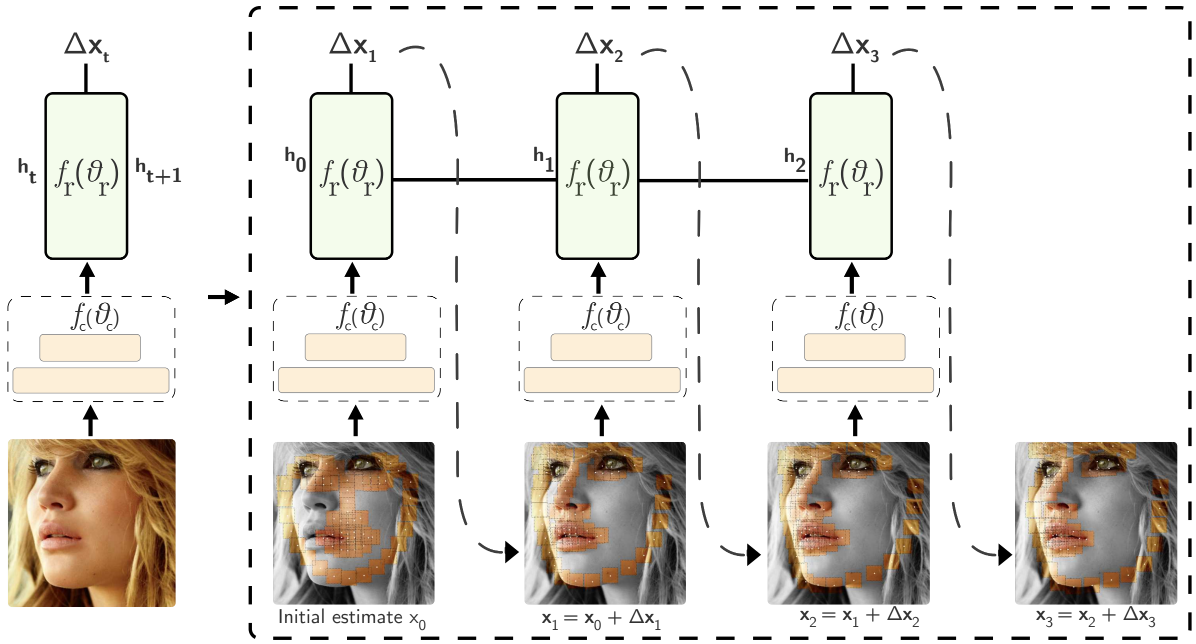

Figure 2: An illustrative example of MDM for a total of $T = 3$ time-steps. Initially the network input consists of a partial image observation, consisting of the patches extracted at the mean face $\mathbf{x}_0$. The extracted patches (30×30) at each time-step are passed through a subsequent convolutional network $f_c(\cdot; \theta_c)$, which in turn produces a representation that is robust to changes in appearance variation. Based on the current state $\mathbf{h}_t$, the mnemonic module (implemented as a recurrent network) generates a new state $\mathbf{h}_{t+1}$ and a new set of descent directions $\Delta\mathbf{x}_{t+1}$ that indicates where the network should focus next. After a total of $T = 3$ time-steps, MDM successfully estimates the landmark locations. An important distinction from the previous work on cascade models [56] is that the weights of the network $\theta = \{\theta_c, \theta_r, \mathbf{x}\}$ are *shared* across time.

where $\mathbf{R}^{(k)} \in \mathbb{R}^{d \times (f+1)}$ and $\mathbf{1}$ is an $f \times 1$ vector of ones that forces the regression matrix to absorb the bias as its final column. Solving for $\mathbf{R}^{(k)}$ reduces to a simple linear least squares problem and is given by $\mathbf{R}^{(k)} = \Delta\mathbf{X}^{(k)}\mathbf{\Phi}^{(k)\dagger}$ where the bias term is concatenated in the design matrix as in Eq. 1 and is thus omitted for brevity. The next cascade step updates the current shape estimate by $\mathbf{x}_i^{(k+1)} = \mathbf{x}_i^* - (\mathbf{x}_i^{(k)} + \Delta\mathbf{x}_i^{(k)})$ and then recomputes the feature matrix using the new shape estimates, $\mathbf{\Phi}^{(k+1)} = [\phi(I_1; \mathbf{x}_1^{(k+1)}), \ldots, \phi(I_n; \mathbf{x}_n^{(k+1)})]$.

4. Mnemonic Descent Method

4.1. Feature extraction

Cascaded regression techniques for face alignment begin with a feature extraction stage, where typically a set of hand-crafted features representing image patches are extracted (e.g., HoG [17], SIFT [32], etc.). The feature extraction stage is required because the images are captured under unconstrained settings and so are likely to contain appearance variations (e.g., in illumination, skin-variations, occlusions etc.), which in turn generate local minima in the energy landscape. The aforementioned representations smooth the landscape in order to minimise the effect of such variations [2]. We note that MDM is feature agnostic and may be straightforwardly used with any such non-linear representations. Nevertheless, although conventional hand-crafted features have proved effective for a multitude of tasks in computer vision [18, 56], this process can still be considered sub-optimal given that these representations are also extracted *independently* of the task-at-hand. The proposed MDM aims to alleviate this issue by means of providing an end-to-end training methodology, in effect jointly discovering both the appropriate non-linear image representations as well as the optimal landmark locations. The feature extraction stage is replaced with a convolutional network module which facilitates learning directional filters leading to the function optimum. Since the training is performed in an end-to-end manner through back-propagation, we essentially learn filters that are used to convolve the image patches *jointly* with the fitting process.

As discussed in the previous section, the proposed algorithm has several advantages over other state-of-the-art algorithms. One of the core contributions is the discovery of the optimal feature representations, since this eliminates the requirement of utilising hand-crafted features which may be sub-optimal. In the next sections, we introduce the MDM algorithm (Sec.4.2) and its end-to-end training in Sec.4.3.

4.2. Model

The main motivation of the proposed MDM is to facilitate smooth convergence by essentially treating the previously *independent* cascade steps as time-steps under a non-linear dynamical system (i.e., modelling dependencies over iterations). Under this paradigm, we essentially learn a single model instead of an independent regressor at each iteration, and by preserving a mnemonic module, MDM enables the steps taken at each iteration to be *dependent* on the previous ones. Effectively, this discourages pitfalls such as missing the function optimum by "stepping-over" it. To this end, we implement the MDM by utilising Recurrent Neural Networks (RNN), which are well-known to be universal approximators for non-linear dynamical systems [42, 22]. In essence, RNNs facilitate feedback connections and thus generate loops and cycles within the network. This enables recurrent networks to account for temporal dependencies arising in the data. In terms of the MDM, this enables modelling dependencies between the iterations and thus the descent directions. Whilst RNNs maintain the topology of feed forward networks (e.g., input, hidden, and output layers), the feedback connections enable the representation of the current *state* of the system which encapsulates the information from the previous inputs. The employment of RNNs has proved highly successful on many applications including machine translation [48], speech recognition [25] and image captioning [53].

In the simplest form, an RNN observes $\mathbf{z}_t$ corresponding to the current time-step t, and based on the previous state $\mathbf{h}_{t-1}$ generates the next hidden state, $\mathbf{h}_t$. Eq. 2 is considered the fundamental equation of a recurrent network (also known as the step function or the recurrence equation)

$$\begin{aligned} \mathbf{h}^{(t+1)} &= f_r(\mathbf{z}^{(t)}, \mathbf{h}^{(t)}; \theta_r) \\ \mathbf{h}^{(t+1)} &= \tanh(\mathbf{W}_{ih}\mathbf{z}^{(t)} + \mathbf{W}_{hh}\mathbf{h}^{(t)}). \end{aligned} \tag{2}$$

Let us now consider the above in the context of cascaded regression, and in particular, in the case of face alignment. The goal of MDM is, given an initial rough estimate of the minimum of the energy landscape, to produce a series of *descent directions* that iteratively lead to the optimum. We denote this initial estimate as $\mathbf{x}^{(0)}$, which for face alignment is commonly a mean face aligned to the output of a face detector [60]. At each time-step t, the mnemonic module partially observes the energy landscape $\mathbf{z}^{(t)}$. Based on this observation, the internal state of MDM is updated accordingly, by adapting the recurrence expression of Eq. 2. Given a new training sample, the recurrence equation becomes,

$$\mathbf{h}^{(t+1)} = \tanh(\mathbf{W}_{hi}\phi(\mathbf{z}; \mathbf{x}^{(t)}) + \mathbf{W}_{hh}\mathbf{h}^{(t)}) \tag{3}$$

where $\mathbf{W}_{hi} \in \mathbb{R}^{f \times u}$ is the hidden-to-input matrix which is used to condition the partial observation of the energy landscape, $\mathbf{W}_{hh} \in \mathbb{R}^{u \times u}$ the hidden-to-hidden matrix which conditions the output of the previous time-step and u corresponds to the dimensionality of the internal state of the recurrent module.

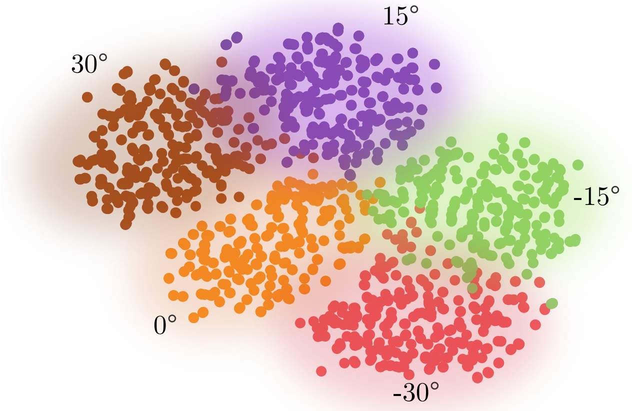

Figure 3: A t-SNE depiction of the internal states ($T = 1$) of MDM when asked to align 2000 randomly selected images of CMU Multi-PIE [24]. Each colour corresponds to a cluster of head pose. This visualisation demonstrates that MDM is effectively partitioning the input data based on the head pose. Best viewed in colour.

During training, the network updates the current shape displacements by projecting the hidden state, which corresponds to the mnemonic element of the algorithm, to the hidden-to-output matrix $\mathbf{W}_{ho} \in \mathbb{R}^{u \times d}$, as

$$\Delta\mathbf{x}^{(t+1)} = \mathbf{x}^{(t)} + \mathbf{W}_{ho}\mathbf{h}^{(t)}. \tag{4}$$

This estimate essentially constitutes the observation for the new time-step and translates to the network observing the local energy landscape around $\mathbf{x}^{(t+1)} = \mathbf{x}^{(t)} + \Delta\mathbf{x}^{(t+1)}$. Note that in this case, the traditional SDM would simply train an independent regressor, and thus fail to utilise the previous states of the algorithm. In fact, in our experimental analysis (c.f. Sec. 5.2) we found, by means of visualisation, that the hidden state of the network at the early stages of the fitting process encapsulates the head pose information. This can then be utilised in subsequent stages to partition the energy landscape and thus enable the model to choose the appropriate descent path to follow. The process is then repeated for a predefined number of time-steps T and is trained by employing Back-propagation Through-Time (BPTT) [54]. In conclusion, the objective function of MDM can be thus defined as[4]

$$\min_{\theta} \|\mathbf{X}^* - \mathbf{X}^{(0)} + \sum_{t=0}^{T-1} \mathbf{W}_{ho}\mathbf{H}^{(t)}\|_F^2 \tag{5}$$

[4] We omit the bias terms for brevity of notation.

where $\mathbf{H}^{(t)} = [\mathbf{h}_1^{(t)}, \ldots, \mathbf{h}_n^{(t)}] \in \mathbb{R}^{u \times n}$ represents the matrix of all states corresponding to each of the n images at time t, and θ all the parameters of the model.

4.3. End-to-end training for face alignment

In Fig. 2, we illustrate the application of MDM for the task of face alignment. Given an initial estimate $\mathbf{x}^{(0)}$, which corresponds to the mean face shape, a collection of patches is extracted and propagated through the convolutional module $f_c(\cdot; \mathbf{x}^{(0)}, \theta_c)$ to obtain the appropriate non-linear feature representations. The recurrent module, $f_r(\cdot; \mathbf{h}^{(0)}, \theta_r)$, then generates the next state $\mathbf{h}^{(1)}$ and the process is repeated for a fixed number of time-steps. We found through experiments on the validation set that unrolling the network for $T = 4$ time steps was sufficient for our task. This is consistent with prior work in cascaded regression where 4 cascade levels are widely deemed sufficient [56]. As the whole network is differentiable end-to-end, we can employ BPTT [54] to learn the parameters of the model.

5. Experimental analysis

In order to evaluate the efficacy of the proposed MDM, we perform rigorous evaluations against the state-of-the-art methods for face alignment (Sec.5.1) where we find a strong improvement against the best performing methods of the 300W competition [39, 38]. In Sec. 5.2 we further examine our model by *(i)* studying the effect of an increasing number of time-steps T, *(ii)* by comparing the outcome of learning a feature extractor (using convolutional features) vs. hand-engineered features (dense-SIFT [32, 52]), and *(iii)* by visualising the internal state of the fitting process, which reveals that it encapsulates the head pose information which the network can employ to partition the space of descent directions in subsequent time-steps.

Datasets. To provide a fair comparison against other recent face alignment methods, we concentrate on the 68-point annotations provided by Sagonas *et al.* [39, 40]. These annotations are provided for 3 existing in-the-wild datasets (LFPW [7], HELEN [30] and AFW [65]) which were originally annotated using different and incompatible markups. Sagonas *et al.* also introduced a new challenging dataset called IBUG [39], also annotated with the 68-point CMU MultiPIE markup [24]. Commonly, these annotations are split into the following subsets: *(i)* the *training set* (3148 images) consisting of LFPW training images (811), HELEN training images (2000) and AFW (337) *(ii)* the *challenging subset* (135) of IBUG (135) *(iii)* the *common subset* (554) of LFPW testing set (224) and HELEN testing set (330) and *(iv)* the *full set* (689) of the union of the common (554) and challenging subsets (135). We do not consider the original annotations of LFPW (29-point markup) or HELEN (194-point markup) as recent works [64, 37, 62] have shown that these databases have become saturated for the original annotations. The above annotations were actually provided as a training/validation set for the 300W face alignment competition [39, 38], which used another set of images strictly for evaluation, called *300W test-set*[5]. The *300W test-set* consists of 600 images split into two subsets, indoor and outdoor, which are said to have been drawn from a similar distribution as the IBUG dataset.

	51-points		68-points	
Method	AUC	Failure (%)	AUC	Failure (%)
ERT [26]	40.60	13.50	32.35	17.00
PO-CR [50]	47.65	11.70	–	–
Chehra [3]	31.12	39.30	–	–
Intraface [56]	38.47	19.70	–	–
Balt. et al. [5]	37.65	17.17	19.55	38.83
Face++ [63]	53.29	5.33	32.81	13.00
Yan et al. [58]	49.07	8.33	34.97	12.67
CFSS [64]	50.79	7.80	39.81	12.30
MDM	**56.34**	**4.20**	**45.32**	**6.80**

Table 1: Quantitative results on the test set of the 300W competition using the AUC (%) and failure rate (calculated at a threshold of 0.08 of the normalised error).

Evaluation. Unfortunately, there is no consistent way of reporting errors for face alignment, even with regards to the common 300W test sets. This is mostly due to variations in error normalisation. To maintain consistency with the results of the 300W competition [39] we use their definition of the interocular distance i.e. the distance between the outer eye corners. We believe that mean errors, particularly without accompanying standard deviations, are not a very informative error metric as they can be highly biased by a low number of very poor fits. Therefore, we provide our evaluation in the form of CED curves, as this is consistent with the results we received from the authors of [39]. We have calculated some further statistics from the CED curves such as the area-under-the-curve (AUC) and the failure rate of each method (we consider any fitting with a point-to-point error greater than 0.08 as a failure). We believe that these are more representative error metrics for the problem of face alignment. We also note that there is a significant difference between 68-point and 49/51-point error metrics due to the inherent difficulty in fitting the boundary points of the face contour. Therefore, where possible, we present both 68 and 49/51 point errors.

Implementation Details. Unless otherwise specified, our network topology consists of two convolutional layers for

[5] Note that the *300W test-set* is different than the *300W full set* commonly used in literature. The former is the test-set used for the 300W competition, which was hidden during the competition and recently made publicly available. The latter refers to the common set of LFPW train, HELEN train and AFW, which was the main training set of the 300W competition. All datasets are available in `http://ibug.doc.ic.ac.uk/resources/facial-point-annotations/`

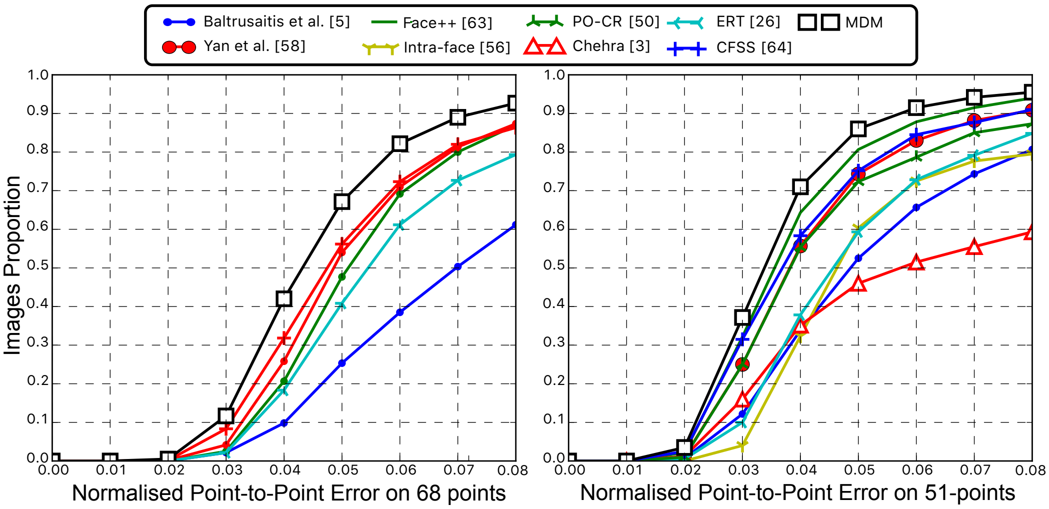

Figure 4: Quantitative results on the test set of the 300W competition (indoor and outdoor combined) [39] for both 68-point (left) and 51-point (right) plots. Only the top 3 performing results from the original competition are shown.

the feature extraction. Each layer employs 32 filters, each with a kernel of 3×3. Each convolutional layer is followed by a rectified linear (ReLU) unit and a 2×2 max-pooling operation. We have added a skip-connection and concatenated the activations from the central crop of the first convolutional layer with the output of the second pooling layer to retain more relevant localisation information that we would otherwise lose from using the max-pooling layers. As a relatively small number of time steps is required to reach convergence we found it sufficient to use a vanilla recurrent module with a state vector of dimensionality 512 units. Finally, a linear projection layer is used to produce the descent directions $\Delta\mathbf{x}^{(t)}$ at each time-step. We provide an example implementation and a pretrained MDM model at `http://trigeorgis.com/mdm`.

For learning the weights of the network we employ stochastic optimisation with Adam [28] with the default hyperparameters, an initial learning rate of 0.001 with exponential decay of 0.95 every $20\,000$ iterations, and minibatch size of 50 images. We choose the best model according to our validation set (300W full set).

For all experiments, our network was trained on the 3148 images of the 300W training set with 68-point markup and the bounding boxes provided by the 300W competition were used for training and testing. Training images were augmented in order to provide extra training data by adding per-pixel Gaussian noise of $\sigma = 0.5$, by mirroring around the vertical axis, and finally with random in-plane rotations $\pm 15°$ generated from a uniform distribution.

5.1. Comparison with State-of-the-art

We compare against state-of-the-art methods in two separate experiments. To provide a relatable benchmark, we evaluated MDM on the full set of 300W. In Fig.6 we provide comparison CED curves against the state-of-the-art methods of Project-Out Cascaded Regression (PO-CR) [50], Coarse-to-fine shape searching (CFSS) [64], Explicit Regression Trees (ERT) [26], Intraface [56, 19], Chehra [3] and a baseline SDM that we implemented using Menpo [1] employing dense-SIFT features. All methods were initialised with the same bounding boxes, as provided by the 300W competition. All methods were chosen due to being publicly available. ERT was re-trained using the implementation provided by DLib [27] with the 300W bounding boxes using the 300W training set. We believe that this experiment demonstrates that the currently available face alignment datasets are becoming saturated, as there is little difference between three of the most recently proposed methods (CFSS, PO-CR, and MDM). Historically, face alignment methods have struggled with not having sufficient training data, and this may have led authors to use the above test-sets as both validation sets (for hyperparameter tuning) and as evaluation sets.

Our primary experiment was evaluated on the test set of the 300W competition [39, 38]. Fig. 4 illustrates the 68-point and 51-point plots of the provided results. The results show that MDM outperforms the rest of the face alignment methods for both the 68-point and 51-point error metrics, setting a new state-of-the-art on the problem of face alignment. It should be noted that the participants of the competition did not have any restrictions on the amount of training data employed which further illustrates the effectiveness of our approach.

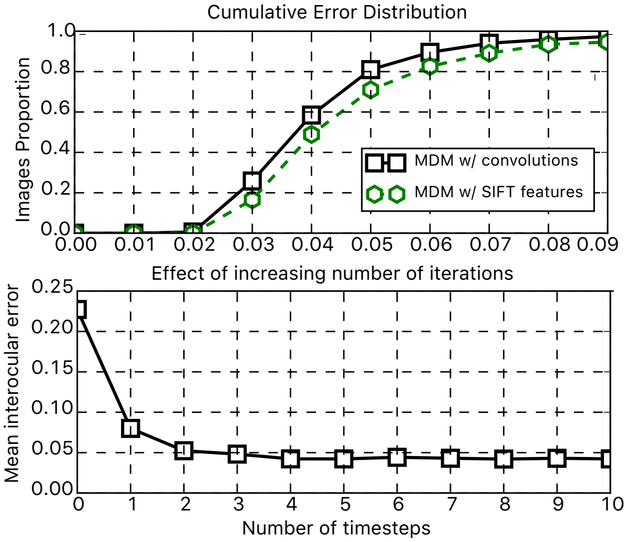

Figure 5: *Top*: Comparison between hand-crafted SIFT features vs. end-to-end CNN features tailored for face alignment. *Bottom*: Increasing the number of time-steps decreases the average error but stabilises after four iterations.

5.2. Self Evaluations

In this section we performed a number of self evaluation experiments to explore the behaviour of our model.

Effect of adding time-steps. In Fig. 5 we show the effect of increasing the number of time-steps for the recurrent network. The mean interocular distance is reported over the whole of the 300W full set. Here we see that only 4 iterations are necessary before the performance plateaus.

Effect of learning features. In Fig. 5 we study the effect of learning features using the CNN in comparison to the SIFT [32] features which are commonly used in many of the cascaded regression algorithms for face alignment [56, 50, 64]. Fig. 5 provides the CED curve on the 300W full set and clearly shows that the learnt features are much more discriminative than the hand-crafted SIFT features.

Partitioning. In Fig.3 we plot a t-SNE [51] visualisation of the $T = 1$ internal states of MDM for 2000 randomly selected images from CMU Multi-PIE [24]. The images were uniformly sampled over a range of out-of-plane head poses in the range $\{-30°, -15°, 0°, 15°, 30°\}$. Fig. 3 clearly shows that MDM is able to partition the space of descent directions according to the head pose. Previously [57, 59], partitioning by pose estimates was considered separately and thus external information was required to perform face alignment. In contrast, MDM naturally learns this partitioning and benefits from improved fitting performance likely due to the implicit clustering of related semantic attributes such as head pose.

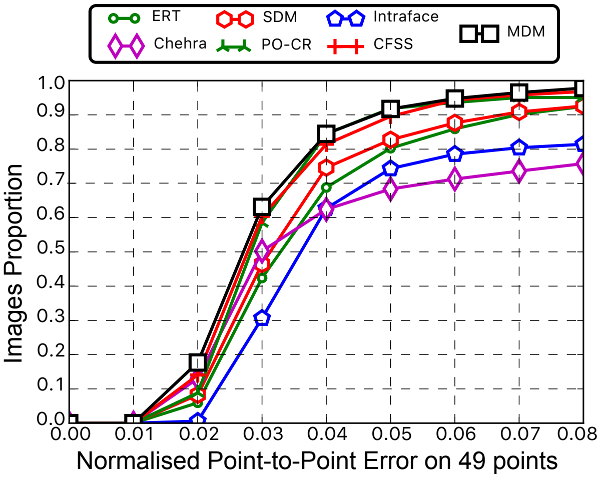

Figure 6: Results on the full testing set of the 300W competition, which was used as a validation set (49-points).

6. Conclusions

We presented the Mnemonic Descent Method (MDM), a non-linear unified model for end-to-end learning of descent directions of non-linear functions. In contrast to existing cascaded regression frameworks, MDM is able to model dependencies between iterations of the cascade by introducing the concept of memory into descent direction learning.

We employ MDM in the area of deformable object alignment. We have proposed the first convolutional recurrent architecture that is able to be trained in an end-to-end manner i.e., from the raw image pixel intensities to the final predictions. By utilising the convolutional module, we have shown that MDM can learn a set of robust features that outperform hand-crafted features for face alignment. Additionally, the recurrent module appears to leverage past information, such as head pose in order to partition the space of descent directions in a data-driven manner. Finally, our approach outperforms the current state-of-the-art for face alignment on the challenging test-set of the 300W competition [39, 38].

Acknowledgements

G. Trigeorgis was supported by EPSRC DTA award at Imperial College London. The work of P. Snape was partially funded by an EPSRC DTA and by the European Community Horizon 2020 [H2020/2014-2020] under grant agreement no. 688520 (TeSLA). The work of E. Antonakos was partially funded by the EPSRC project EP/J017787/1 (4D-FAB). The work of S. Zafeiriou was funded by the FiDiPro program of Tekes (project number: 1849/31/2015). We thank the NVIDIA Corporation for donating a Tesla K40 GPU used in this work.

References

[1] J. Alabort-i-Medina, E. Antonakos, J. Booth, P. Snape, and S. Zafeiriou. Menpo: A comprehensive platform for parametric image alignment and visual deformable models. In *Proceedings of the ACM International Conference on Multimedia*, MM '14, pages 679–682, New York, NY, USA, 2014. ACM.

[2] E. Antonakos, J. Alabort-i-Medina, G. Tzimiropoulos, and S. Zafeiriou. Feature-based lucas–kanade and active appearance models. *IEEE Transactions on Image Processing*, 24(9):2617–2632, 2015.

[3] A. Asthana, S. Zafeiriou, S. Cheng, and M. Pantic. Incremental face alignment in the wild. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1859–1866. IEEE, 2014.

[4] A. Asthana, S. Zafeiriou, G. Tzimiropoulos, S. Cheng, and M. Pantic. From pixels to response maps: Discriminative image filtering for face alignment in the wild. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2015.

[5] T. Baltrusaitis, P. Robinson, and L.-P. Morency. Constrained local neural fields for robust facial landmark detection in the wild. In *International Conference on Computer Vision Workshops (ICCVW)*, pages 354–361. IEEE, 2013.

[6] P. N. Belhumeur, D. W. Jacobs, D. J. Kriegman, and N. Kumar. Localizing parts of faces using a consensus of exemplars. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(12):2930–2940, 2013.

[7] P. N. Belhumeur, D. W. Jacobs, D. J. Kriegman, and N. Kumar. Localizing parts of faces using a consensus of exemplars. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35:2930–2940, 2013.

[8] V. Blanz and T. Vetter. In the proceedings of siggraph. In *Computer Graphics and Interactive Techniques*, pages 187–194. ACM Press/Addison-Wesley Publishing Co., 1999.

[9] A. M. Buchanan and A. W. Fitzgibbon. Damped Newton Algorithms for Matrix Factorization with Missing Data. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 2, pages 316–322. IEEE, 2005.

[10] X. P. Burgos-Artizzu, P. Perona, and P. Dollár. Robust face landmark estimation under occlusion. In *International Conference on Computer Vision (ICCV)*, pages 1513–1520. IEEE, 2013.

[11] X. Cao, Y. Wei, F. Wen, and J. Sun. Face alignment by explicit shape regression. *International Journal of Computer Vision*, 107(2):177–190, 2014.

[12] X. Chen and C. L. Zitnick. Learning a recurrent visual representation for image caption generation. *arXiv preprint arXiv:1411.5654*, 2015.

[13] T. F. Cootes, G. J. Edwards, and C. J. Taylor. Active appearance models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(6):681–685, 2001.

[14] T. F. Cootes, M. C. Ionita, C. Lindner, and P. Sauer. Robust and Accurate Shape Model Matching Using Random Forest Voting. In *European Conference on Computer Vision (ECCV)*, pages 278–291. Springer, 2012.

[15] T. F. Cootes and P. Kittipanya-ngam. Comparing variations on the active appearance model algorithm. In *British Machine Vision Conference*, pages 1–10. Citeseer, 2002.

[16] D. Cristinacce and T. Cootes. Automatic feature localisation with constrained local models. *Pattern Recognition*, 41(10):3054–3067, 2008.

[17] N. Dalal and B. Triggs. Histograms of Oriented Gradients for Human Detection. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages 886–893. IEEE, 2005.

[18] N. Dalal, B. Triggs, and C. Schmid. Human detection using oriented histograms of flow and appearance. In *European Conference on Computer Vision*, pages 428–441. Springer, 2006.

[19] F. De la Torre, W.-S. Chu, X. Xiong, F. Vicente, X. Ding, and J. F. Cohn. Intraface. In *Automatic Face and Gesture Recognition*, 2015.

[20] P. Dollár, P. Welinder, and P. Perona. Cascaded Pose Regression. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1078–1085. IEEE, 2010.

[21] J. Donahue, L. Anne Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell. Long-Term Recurrent Convolutional Networks for Visual Recognition and Description. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2625–2634, 2015.

[22] K.-i. Funahashi and Y. Nakamura. Approximation of dynamical systems by continuous time recurrent neural networks. *Neural networks*, 6(6):801–806, 1993.

[23] R. Garg, A. Roussos, and L. Agapito. Dense variational reconstruction of non-rigid surfaces from monocular video. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1272–1279, 2013.

[24] R. Gross, I. Matthews, J. Cohn, T. Kanade, and S. Baker. Multi-Pie. *Image and Vision Computing*, 28(5):807–813, 2010.

[25] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A.-r. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath, et al. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *Signal Processing Magazine*, 29(6):82–97, 2012.

[26] V. Kazemi and J. Sullivan. One Millisecond Face Alignment with an Ensemble of Regression Trees. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1867–1874. IEEE, 2014.

[27] D. E. King. Dlib-ml: A machine learning toolkit. *Journal of Machine Learning Research*, 10:1755–1758, 2009.

[28] D. Kingma and J. Ba. Adam: A method for stochastic optimization. *International Conference on Learning Representations*, 2014.

[29] V. Le, J. Brandt, Z. Lin, L. Bourdev, and T. S. Huang. Interactive facial feature localization. In *European Conference on Computer Vision (ECCV)*, pages 679–692. Springer, 2012.

[30] V. Le, J. Brandt, Z. Lin, L. Bourdev, and T. S. Huang. Interactive facial feature localization. In *Computer Vision–ECCV 2012*, pages 679–692. Springer, 2012.

[31] D. Lee, H. Park, and C. D. Yoo. Face Alignment using Cascade Gaussian Process Regression Trees. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4204–4212, 2015.

[32] D. G. Lowe. Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[33] B. D. Lucas, T. Kanade, et al. An iterative image registration technique with an application to stereo vision. In *International Joint Conference on Artificial Intelligence*, volume 81, pages 674–679, 1981.

[34] P. Martins, R. Caseiro, J. F. Henriques, and J. Batista. Discriminative bayesian active shape models. In *European Conference on Computer Vision (ECCV)*, pages 57–70. Springer, 2012.

[35] I. Matthews and S. Baker. Active appearance models revisited. *International Journal of Computer Vision*, 60(2):135–164, 2004.

[36] P. H. Pinheiro and R. Collobert. Recurrent Convolutional Neural Networks for Scene Labeling. In *International Conference on Machine Learning*, pages 2625–2634, 2014.

[37] S. Ren, X. Cao, Y. Wei, and J. Sun. Face alignment at 3000 fps via regressing local binary features. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1685–1692. IEEE, 2014.

[38] C. Sagonas, E. Antonakos, G. Tzimiropoulos, S. Zafeiriou, and M. Pantic. 300 faces in-the-wild challenge: Database and results. *Image and Vision Computing, Special Issue on Facial Landmark Localisation "In-The-Wild"*, 2016.

[39] C. Sagonas, G. Tzimiropoulos, S. Zafeiriou, and M. Pantic. 300 faces in-the-wild challenge: The first facial landmark localization challenge. In *International Conference on Computer Vision Workshops (ICCVW)*, pages 397–403. IEEE, 2013.

[40] C. Sagonas, G. Tzimiropoulos, S. Zafeiriou, and M. Pantic. A semi-automatic methodology for facial landmark annotation. In *International Conference on Computer Vision and Pattern Recognition Workshops (CVPR-W)*, pages 896–903, 2013.

[41] J. M. Saragih, S. Lucey, and J. F. Cohn. Deformable model fitting by regularized landmark mean-shift. *International Journal of Computer Vision*, 91(2):200–215, 2011.

[42] J. Schmidhuber. Deep learning in neural networks: An overview. *Neural Networks*, 61:85 – 117, 2015.

[43] B. Shi, X. Bai, W. Liu, and J. Wang. Deep Regression for Face Alignment. *arXiv preprint arXiv:1409.5230*, 2014.

[44] C. C. Slama, C. Theurer, S. W. Henriksen, et al. *Manual of photogrammetry*. Number Ed. 4. American Society of photogrammetry, 1980.

[45] P. Snape, A. Roussos, Y. Panagakis, and S. Zafeiriou. Face flow. In *International Conference on Computer Vision (ICCV)*, pages 2993–3001. IEEE, 2015.

[46] D. Sun, S. Roth, and M. J. Black. A quantitative analysis of current practices in optical flow estimation and the principles behind them. *International Journal of Computer Vision*, 106(2):115–137, 2014.

[47] Y. Sun, X. Wang, and X. Tang. Deep Convolutional Network Cascade for Facial Point Detection. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3476–3483. IEEE, 2013.

[48] I. Sutskever, O. Vinyals, and Q. V. Le. Sequence to Sequence Learning with Neural Networks. In *Advances in Neural Information Processing Systems*, pages 3104–3112, 2014.

[49] P. A. Tresadern, M. C. Ionita, and T. F. Cootes. Real-Time Facial Feature Tracking on a Mobile Device. *International Journal of Computer Vision*, 96(3):280–289, 2012.

[50] G. Tzimiropoulos. Project-Out Cascaded Regression With an Application to Face Alignment. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3659–3667, 2015.

[51] L. Van der Maaten and G. Hinton. Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9(2579-2605):85, 2008.

[52] A. Vedaldi and B. Fulkerson. VLFeat: An open and portable library of computer vision algorithms. `http://www.vlfeat.org/`, 2008.

[53] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan. Show and Tell: A Neural Image Caption Generator. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3156–3164, June 2015.

[54] P. J. Werbos. Backpropagation Through Time: what it does and how to do it. *Proceedings of the IEEE*, 78(10):1550–1560, 1990.

[55] Y. Wu, Z. Wang, and Q. Ji. Facial Feature Tracking Under Varying Facial Expressions and Face Poses Based on Restricted Boltzmann Machines. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3452–3459. IEEE, 2013.

[56] X. Xiong and F. De la Torre. Supervised descent method and its applications to face alignment. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 532–539. IEEE, 2013.

[57] X. Xiong and F. De la Torre. Global Supervised Descent Method. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2664–2673, 2015.

[58] J. Yan, Z. Lei, D. Yi, and S. Z. Li. Learn to combine multiple hypotheses for accurate face alignment. In *International Conference on Computer Vision Workshops (ICCVW)*, pages 392–396. IEEE, 2013.

[59] H. Yang, W. Mou, Y. Zhang, I. Patras, H. Gunes, and P. Robinson. Face Alignment Assisted by Head Pose Estimation. In *British Machine Vision Conference*, 2015.

[60] S. Zafeiriou, C. Zhang, and Z. Zhang. A survey on face detection in the wild: past, present and future. *Computer Vision and Image Understanding*, 138:1–24, 2015.

[61] J. Zhang, S. Shan, M. Kan, and X. Chen. Coarse-to-fine auto- encoder networks (CFAN) for real-time face alignment. In *European Conference on Computer Vision*, pages 1–16. Springer, 2014.

[62] Z. Zhang, P. Luo, C. C. Loy, and X. Tang. Learning Deep Representation for Face Alignment with Auxiliary Attributes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2015.

[63] E. Zhou, H. Fan, Z. Cao, Y. Jiang, and Q. Yin. Extensive facial landmark localization with coarse-to-fine convolutional network cascade. In *International Conference on Computer Vision Workshops (ICCVW)*, pages 386–391. IEEE, 2013.

[64] S. Zhu, C. Li, C. C. Loy, and X. Tang. Face Alignment by Coarse-to-Fine Shape Searching. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4998–5006, 2015.

[65] X. Zhu and D. Ramanan. Face detection, pose estimation, and landmark localization in the wild. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2879–2886, 2012.

[66] D. Zikic, A. Kamen, and N. Navab. Revisiting Horn and Schunck: Interpretation as Gauss-Newton Optimisation. In *British Machine Vision Conference*, pages 1–12, 2010.