SIMULTANEOUS CASCADED REGRESSION

Pedro Martins, Jorge Batista

Institute of Systems and Robotics, University of Coimbra, Portugal

ABSTRACT

This paper addresses the problem of localizing facial landmarks with deformable face models using cascaded regression strategies. Recently, these methods have become quite popular, standing out as simple and efficient approaches to optimize nonlinear objective functions. In this paper, we target the well-known Lucas and Kanade (LK) image alignment formulation and introduce the Simultaneous Cascaded Regression (SCR) technique, which can be considered as a cascaded regression extension of the Simultaneous Forwards Additive / Inverse Composition approaches. In contrast to previous LK techniques (Newton based optimizations) which require to recompute Jacobian and Hessians matrices at each iteration, our approach learns (offline) a sequence of descent directions, effectively behaving as averaged steepest descent matrices. Under this revised technique, we propose a part-based generative model (with a linear warp function), that accounts with the underlying shape and appearance structure embedded into regression process itself. Our method is validated on a number of experiments on several datasets (LFPW, LFW, HELEN, 300W), demonstrating a noticeable gain in accuracy/fitting performance when compared with other face alignment solutions.

Index Terms— Non-rigid face registration, face alignment, deformable models, facial feature localization, cascaded regression.

1. INTRODUCTION

Nonrigid deformable face alignment (registration) plays a fundamental role in a wide range of computer vision applications. Examples include visual tracking, face recognition, 3D head pose estimation, video encoding, etc. In general, the face alignment task seeks to accurately locate the set of landmark keypoints (p.e. eyes corners, nose nostrils, mouth, eyebrows) that defines its detailed structure.

One of the most popular, long-standing, technique is the Active Appearance Model (AAM) [1] [2] [3] [4]. Briefly, the AAM combine generative models of both shape and appearance (texture) that allow efficient deformable matching of unseen instances. Fitting such a model can be posed as a nonlinear optimization that finds, in some sense, the best set of shape and appearance parameters that minimizes the difference between an image and the model itself.

Over the years, several optimization strategies have been proposed. The original formulation [1] relies on a fixed regression approach (learning the relation between the appearance error and the optimal parameters). Afterwards, extended approaches included adaptive linear regression [5] [6] and boosted regression [7] [8].

Later, optimization driven strategies were introduced, in which the Lucas and Kanade (LK) framework [9] [10] played a fundamental role, as it offered the possibility of using Newton methods with analytical gradients. Under this strategy, probably the most popular is the efficient Project-Out (PO) [2] [11] algorithm based on the Inverse Compositional (IC) [12] update scheme. In contrast to the standard Forwards Additive technique, the IC reformulates the optimization, by reversing the role between the image and the model. In this setting, the Jacobian and the Hessian matrices become constant and hence can be precomputed (note that the PO is a technique that removes the effects of the appearance basis from the Jacobian). In essence, the PO-IC was designed to be fast, however it lacked in accuracy, specially under unseen appearances. This was addressed later by the far more accurate Simultaneous Inverse Compositional (SIC) [11] [13], although at the cost of a much higher computational burden. Still, efficient versions of SIC algorithm [14] [3] were also proposed. Other solutions were also pursued, namely robust extensions into the Fourier domain [15], adding 3D shape priors [16] [17] and nonlinear feature representations [18].

The previous mentioned techniques, which are all based in piecewise affine warps, have full holistic appearance representations (i.e. all pixels belonging into the face are being modelled). However, modern enhanced methods have improved representations by using a part-based model (accounting for local features around each landmark). As example, we highlight the Constrained Local Model (CLM) [19] [20] [21] [22] [23] and the Deformable Part Model (DPM) [24] [25]. It is worth mentioning that all these approaches, except [25], have discriminative based appearances.

Recently, a new paradigm has emerged, the cascade regression approach [26] [27] [28] [29] [30] [31] [32] [33]. These techniques allow to optimize nonlinear objective functions very efficiently, by learning (offline) a set of descent directions. Note that this process is divided into small steps, by learning an ensemble of regressors (chained in series). Model fitting simply relies in applying the regressors recursively (where each regressor relates the current features with the updates to be made to the parameters). Broadly speaking, these methods differ from each other by the way as the regression process is made, p.e. boosted regression [26] [34] [29] [30], least-squares regression [27] or Gaussian Processes regression [31].

In this paper, we revisit the Lucas and Kanade image alignment framework and introduce a cascaded regression extension of the Simultaneous algorithm, designed here as the Simultaneous Cascaded Regression (SCR). Like in its LK counterpart, the SCR optimizes shape, pose and appearance parameters at together, however, it uses a part-based appearance representation and its objective function is formulated in terms of a sequence of regressions (ridge regression). Our approach is closely related to the Supervised Descent Method (SDM) [27] and to the Project-Out Cascade Regression (PO-CR) [28] where the former simply attempts to estimate a general regression matrix and the latter uses the PO algorithm to discard the appearance effects from the optimization. Our SCR approach differs from the previous in the way it embeds the full model structure into the regression process. Deep within the formulation (and opposed to SDM) our approach has some computational advantages: it does not require a low dimensional reduction step (which is the SDM's learning bottleneck) and it does not require to invert the huge data

This work was supported by the Portuguese Science Foundation (FCT) through the grant SFRH/BPD/90200/2012 (Pedro Martins).

matrix that hold all the accumulated samples for regression. In contrast to PO-CR, our approach is considerable more accurate while still maintaining a high degree of computational efficiency.

2. PART-BASED PARAMETRIC MODEL

This section defines the generative part-based model (local appearance patches regularized by a linear shape constraint) and describes the standard fitting approach (Lucas & Kanade based optimization).

2.1. Shape and Appearance Models

A 2D shape with v landmarks, or fiducials, is represented by the vector $\mathbf{s} = (x_1, \ldots, x_v, y_1, \ldots, y_v)^T \in \mathbb{R}^{2v}$. Typically, the shape model is captured from a set annotated examples. Afterwards a Procrustes analysis is applied in each training example removing similarity effects. The shape model itself arise from applying a Principal Components Analysis (PCA) onto the set of normalized shapes. The resulting motion model, defined here by the warp function $\mathcal{W}$, encodes the shape deformation combined with 2D pose as

$$\mathcal{W}(\mathbf{s};\mathbf{p}) = \mathbf{s}_0 + \sum_{i=1}^{n+4} \phi_i p_i = \mathbf{s}_0 + \Phi\mathbf{p} \tag{1}$$

where $\mathbf{p} \in \mathbb{R}^{n+4}$ is the shape parameters vector (where the first n elements represent the deformation weights and the last four the 2D pose), Φ holds $n+4$ eigenvectors of the shape subspace with the last four being a special set, defined as function of the mean shape $\mathbf{s}_0$ (see [2]). The mentioned pose parameters follow the reparameterization $[s\cos(\theta) - 1, s\sin(\theta), t_x, t_y]^T$ where s is the scale, θ the rotation and (t_x, t_y) are 2D translations (defined w.r.t. $\mathbf{s}_0$).

The appearance model $\mathcal{A}(\mathbf{x};\boldsymbol{\lambda})$ also consists of a linear representation (EigenFaces) given by

$$\mathcal{A}(\mathbf{x};\boldsymbol{\lambda}) = \mathbf{A}_0(\mathbf{x}) + \sum_{i=1}^{m} \mathbf{A}_i(\mathbf{x})\lambda_i = \mathbf{A}_0 + \mathbf{A}\boldsymbol{\lambda} \tag{2}$$

where each appearance instance $\mathcal{A}(\mathbf{x};\boldsymbol{\lambda})$, can be expressed by a base appearance $\mathbf{A}_0$ (mean appearance) plus a linear combination of m appearance images $\mathbf{A}_i$ (the EigenFaces), the $\boldsymbol{\lambda} \in \mathbb{R}^m$ is the appearance parameters that, once again, define the mixing weights. In this work, a multi-dimensional feature representation (HoG [35]) is used.

The vector $\mathbf{x}$ represents the set of pixel locations where the appearance is defined. In this case, a local patch based appearance (around each landmark) is used

$$\mathbf{x} = \bigcup_{i=1}^{v} \Omega_{\mathbf{s}_i} \tag{3}$$

where $\Omega_{\mathbf{s}_i}$ denotes a $L \times L$ squared support region around the center location $\mathbf{s}_i$ (which in turn is generated by the warp function in eq. 1). The figure 1 shows a visual representation of this local appearance model. For the next sections, we drop the spatial dependence on $\mathbf{x}$, and use only the condensed representation shown in the eq. 2.

2.2. Simultaneous Forwards Additive (SFA)

One of the most universal LK optimizations is the Simultaneous Forwards Additive (SFA). The SFA optimization goal seeks to find the shape and appearance parameters that minimizes the difference between the model and the sampled target image. Formally, we can write

$$\arg\min_{\mathbf{p},\boldsymbol{\lambda}} \|\mathbf{A}_0 + \mathbf{A}\boldsymbol{\lambda} - \mathbf{I}(\mathcal{W}(\mathbf{p}))\|^2 \tag{4}$$

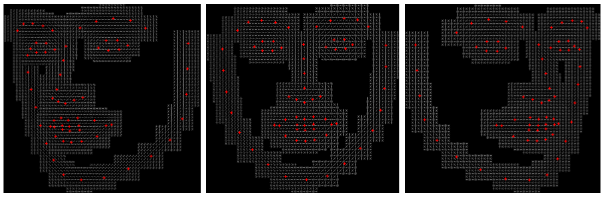


Fig. 1. Visual representation of the generative part based model. The warp function $\mathcal{W}(\mathbf{s};\mathbf{p})$ defines the shape's landmark localization $\mathbf{s}$ thought the parameters $\mathbf{p}$. Similarly, the appearance model $\mathcal{A}(\mathbf{x};\boldsymbol{\lambda})$ synthesise the local patch features (HoG [35]) by the parameters $\boldsymbol{\lambda}$.

where $\mathbf{I}(\mathcal{W}(\mathbf{p}))$ represents the local feature extraction of the input image sampled at the location $\mathcal{W}(\mathbf{p})$ which is governed by eq. 1. Note that, even in the face of such linear models, this optimization is highly nonlinear, mainly because there is no direct correlation between the face appearance and its global localization.

A possible way to solve optimization 4 is to apply the LK image alignment framework [10] [11], by iteratively solving for small additive updates

$$\arg\min_{\Delta\mathbf{p},\Delta\boldsymbol{\lambda}} \|\mathbf{A}_0 + \mathbf{A}(\boldsymbol{\lambda} + \Delta\boldsymbol{\lambda}) - \mathbf{I}(\mathcal{W}(\mathbf{p} + \Delta\mathbf{p}))\|^2 . \tag{5}$$

A first-order Taylor expansion in eq. 5 results in

$$\arg\min_{\Delta\mathbf{p},\Delta\boldsymbol{\lambda}} \|\mathbf{A}_0 + \mathbf{A}\boldsymbol{\lambda} + \mathbf{A}\Delta\boldsymbol{\lambda} - \mathbf{I}(\mathcal{W}(\mathbf{p})) - \mathbf{J}_\mathbf{p}\Delta\mathbf{p}\|^2 \tag{6}$$

where the Jacobian term $\mathbf{J}_\mathbf{p} = \nabla\mathbf{I}\frac{\partial\mathcal{W}(\mathbf{p})}{\partial\mathbf{p}}$ covers the term $\nabla\mathbf{I} = \left(\frac{\partial\mathbf{I}}{\partial x}, \frac{\partial\mathbf{I}}{\partial y}\right)^T$ being the x-y gradients evaluated at the image frame and $\frac{\partial\mathcal{W}(\mathbf{s}_k;\mathbf{p})}{\partial\mathbf{p}_i} = \phi_i^k$ is the Jacobian of the Warp function evaluated at $\mathbf{p}$ (with $k = 1, \ldots, v$ landmarks, $i = 1, \ldots, n+4$ parameters). We remark that, for this particular warp, the $\mathcal{W}$ function (eq. 1) and its Jacobian are constant.

The solution of eq. 6 takes the form of

$$\begin{bmatrix} \Delta\mathbf{p} \\ \Delta\boldsymbol{\lambda} \end{bmatrix} = \mathbf{H}_{\mathbf{FA}}^{-1}\mathbf{J}_{\mathbf{FA}}^T\left[\mathbf{A}_0 + \mathbf{A}\boldsymbol{\lambda} - \mathbf{I}(\mathcal{W}(\mathbf{p}))\right] \tag{7}$$

where $\mathbf{J}_{\mathbf{FA}} = [\mathbf{J}_\mathbf{p}, \mathbf{A}]$ is the extended Jacobian and $\mathbf{H}_{\mathbf{FA}} = \mathbf{J}_{\mathbf{FA}}^T\mathbf{J}_{\mathbf{FA}}$ is the Gauss-Newton approximation to the Hessian matrix. This solution actually leads to a rather computationally expensive process where both the Jacobian and the Hessian needed to be recomputed every iteration. Finally, the shape and appearance parameters are updated as $\mathbf{p} \leftarrow \mathbf{p} + \Delta\mathbf{p}$ and $\boldsymbol{\lambda} \leftarrow \boldsymbol{\lambda} + \Delta\boldsymbol{\lambda}$, respectively.

2.3. Simultaneous Inverse Compositional (SIC)

The Inverse Compositional (IC) [12] strategy was initially designed to reduce the computational burden of the alignment by reformulating the optimization 4 in terms of a (inverse) compositional update. This was accomplished by inverting the roles between the model and the image (w.r.t. the linearization). According, the optimization in 5 is converted into

$$\arg\min_{\Delta\mathbf{p},\Delta\boldsymbol{\lambda}} \|\mathbf{A}_0(\mathcal{W}(\Delta\mathbf{p})) + \mathbf{A}(\mathcal{W}(\Delta\mathbf{p}))(\boldsymbol{\lambda} + \Delta\boldsymbol{\lambda}) - \mathbf{I}(\mathcal{W}(\mathbf{p}))\|^2 . \tag{8}$$

Following a similar procedure, the least-squares solution comes as

$$\begin{bmatrix} \Delta\mathbf{p} \\ \Delta\boldsymbol{\lambda} \end{bmatrix} = -\mathbf{H}_{\mathbf{IC}}^{-1}\mathbf{J}_{\mathbf{IC}}^T\left[\mathbf{A}_0 + \mathbf{A}\boldsymbol{\lambda} - \mathbf{I}(\mathcal{W}(\mathbf{p}))\right] \tag{9}$$

where now the Hessian is $\mathbf{H_{IC}} = \mathbf{J_{IC}^{T}J_{IC}}$ and the overall (expanded) Jacobian follows $\mathbf{J_{IC}} = \left((\nabla\mathbf{A}_0 + \nabla\mathbf{A}\boldsymbol{\lambda}) \frac{\partial\mathcal{W}(\mathbf{0})}{\partial\mathbf{p}},\ \mathbf{A} \right)$ with most of the terms being precomputed. Note that, the Jacobian terms are now expressed w.r.t. the model. The IC shape parameters update, for this warp, $\mathcal{W}(\mathbf{s},\mathbf{p}) \leftarrow \mathcal{W}(\mathbf{s},\mathbf{p}) \circ \mathcal{W}(\mathbf{s},\Delta\mathbf{p})^{-1}$ reduces to $\mathbf{p} \leftarrow \mathbf{p} - \Delta\mathbf{p}$.

3. SIMULTANEOUS CASCADED REGRESSION (SCR)

The cascaded regression framework seeks to learn a succession of regression matrices, defined as $\{\mathbf{R}^k\}_1^K$, that follow the sequence

$$\mathbf{r}^k = \mathbf{r}^{k-1} + \mathbf{R}^{k-1}\left(\mathbf{f}(\mathbf{r}^{k-1}) - \mathbf{f}(\mathbf{r}_*)\right), \quad k = 1,\dots,K \qquad (10)$$

where $\mathbf{r} = [\mathbf{p}|\boldsymbol{\lambda}] \in \mathbb{R}^{n+4+m}$ represents the latent parameters (both shape and appearance parameters concatenated together), the $\mathbf{r}^0$ is the initial parameters estimate (usually obtained from the output of a face detection), the $\mathbf{r}_*$ is the ground truth parameters (derived from the image annotations), the $\mathbf{f}(\mathbf{r}) \equiv \mathbf{I}(\mathcal{W}(\mathbf{r}))$ stands for the local image features extracted at the locations generated by the model's parameters and K is the total number of cascade levels.

Instead of attempting to estimate a generic regression matrix (as in SDM [27]), the complexity of the problem can be reduced by including some structure knowledge into the objective function. Following the previous SFA optimization in 5, we can firstly estimate the average Jacobian $\mathbf{J}_{\mathbf{S}}^k$ across the full set of examples and under multiple possible initializations

$$\arg\min_{\mathbf{J}_{\mathbf{S}}^k} \sum_{i=1}^{N} \int p(\mathbf{r}') \left\| \mathbf{A}_0 + \mathbf{A}\boldsymbol{\lambda}_i^k + \mathbf{J}_{\mathbf{S}}^k \Delta\mathbf{r}_i^k - \mathbf{I}_i(\mathcal{W}(\mathbf{p}_i^k)) \right\|^2 \partial\mathbf{r}' \qquad (11)$$

where the index i refers to the i^{th} training image (existing N in total) and k is the current cascade level. The $\Delta\mathbf{r}_i^k = \begin{bmatrix} \mathbf{p}_i^k - \mathbf{p}_* \\ \boldsymbol{\lambda}_i^k - \boldsymbol{\lambda}_* \end{bmatrix}$ stands for the combined disturbed parameters deviation from the ground truth. Note that the second and last terms of eq. 11 are also affected by the parameters disturbance.

In the previous, assuming that $\mathbf{r}'$ is drawn from a Normal distribution $\mathcal{N}(\mathbf{0},\Sigma_{\mathbf{r}})$, the optimization in 11 can be approximated by the discrete form

$$\arg\min_{\mathbf{J}_{\mathbf{S}}^k} \sum_{i=1}^{N}\sum_{j=1}^{M} \left\| \mathbf{A}_0 + \mathbf{A}\boldsymbol{\lambda}_{ij}^k + \mathbf{J}_{\mathbf{S}}^k \Delta\mathbf{r}_{ij}^k - \mathbf{I}_i(\mathcal{W}(\mathbf{p}_{ij}^k)) \right\|^2 \qquad (12)$$

where the double indexes (i,j) refer to the j^{th} perturbation (M in total) with respect to the i^{th} image.

A solution for the optimization in 12 can be found by Ridge Regression, which is given by

$$\mathbf{J}_{\mathbf{S}}^k = \left(\Delta\mathbf{r}\Delta\mathbf{r}^T + \lambda_1\mathbf{I}_d\right)^{-1} \Delta\mathbf{r}\,\mathbf{E}^T \qquad (13)$$

where $\mathbf{E}$ is a large data matrix holding (by columns) the residual feature error between the sampled image and the appearance model for each perturbations at each image, or $\mathbf{E}_{ij} = \mathbf{I}_i(\mathcal{W}(\mathbf{p}_{ij}^k)) - \mathbf{A}_0 - \mathbf{A}\boldsymbol{\lambda}_{ij}^k$. Note that, despite the notation includes two indexes, in practice this could be implemented with simple indexation: idx $= (i-1)\times M + j$. The matrix $\Delta\mathbf{r}$ contains all parameters deviations (also stacked by columns), λ_1 is a regularization parameter (preventing overfitting) and $\mathbf{I}_d$ is a $d = n+4+m$ dimensional identity matrix. It is worth mentioning that, in contrast to other methods (namely SDM [27]), the data matrix that holds all features appears at the right size of eq. 13, which avoids the computation of inverting a large matrix.

```
Learn the shape and appearance models $(\mathbf{s}_0, \Phi, \mathbf{A}_0, \mathbf{A})$
Get an initial estimate for all virtual instances $\mathbf{r}_{ij}$
for cascade $k = 1$ to $K$ do
    $\Sigma_{\mathbf{r}} = \text{cov}(\mathbf{r}_{ij} - \mathbf{r}_*)$            //Estimate noise
    for image $i = 1$ to $N$ do
        for perturbation $j = 1$ to $M$ do
            $\mathbf{r}_{ij} = \mathbf{r}_{ij} + \nu, \quad \nu \sim \mathcal{N}(\mathbf{0},\Sigma_{\mathbf{r}})$ //Add noise
            $\Delta\mathbf{r}_{ij} = \mathbf{r}_{ij} - \mathbf{r}_*$   //Deviation from GT
            $\mathbf{I}_i(.) \rightarrow \mathcal{S}^{-1}(\mathbf{I}_i(.), \mathbf{p}_{ij})$      //Warp image
            $\mathbf{I}_i(\mathcal{W}(\mathbf{p}_{ij}))$           //Extract features
            $\mathbf{E}_{ij} = \mathbf{I}_i(\mathcal{W}(\mathbf{p}_{ij})) - \mathbf{A}_0 - \mathbf{A}\boldsymbol{\lambda}_{ij}$
        end
        Estimate the Jacobian $\mathbf{J}_{\mathbf{S}}^k$ (eq. 13)
        Compute the update matrix $\mathbf{R}^k$ (eq. 14)
    end
    $\mathbf{r}^{k+1} \leftarrow \mathbf{r}^k + \mathbf{R}^k\,\mathbf{E}$             //Cascade update
end
```

Algorithm 1: Simultaneous Cascade Regression learning.

Regarding the Hessian matrix, such matrix can be computed using the Gauss-Newton approximation ($\mathbf{H_S} = \mathbf{J_S^T J_S}$). Although, from the numerical stability point of view, such estimate can be improved by adding a small regularization weight (λ_2). According the overall update matrix for the k^{th} cascade iteration is given by

$$\mathbf{R}^k = \left(\mathbf{H}_{\mathbf{S}}^k + \lambda_2\mathbf{I}_d\right)^{-1} \mathbf{J}_{\mathbf{S}}^{k\ T}. \qquad (14)$$

Finally, the cascade update becomes

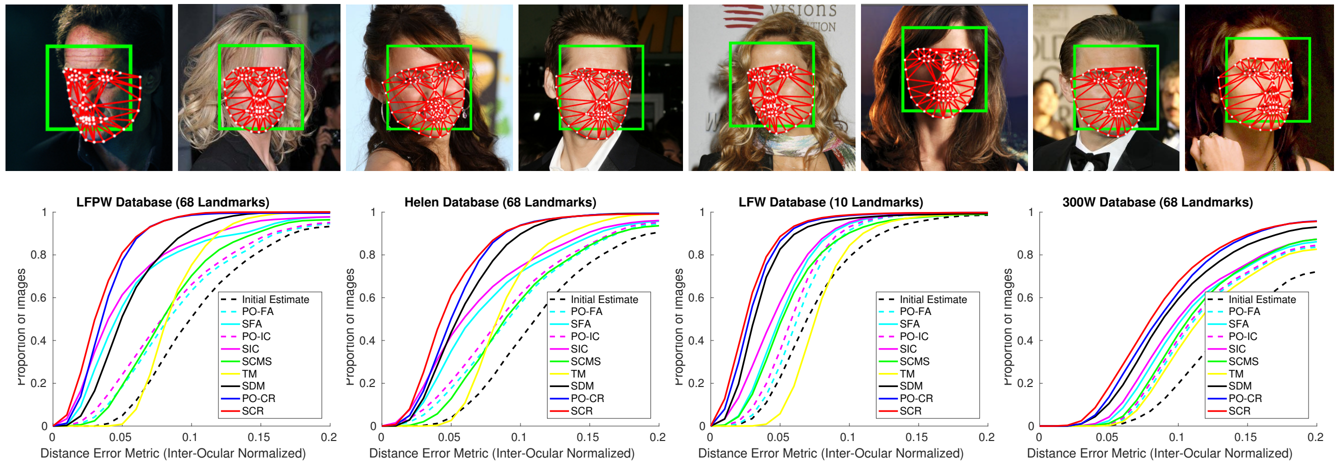
$$\Delta\mathbf{r}^k = \mathbf{R}^k\left(\mathbf{I}(\mathcal{W}(\mathbf{p}^k)) - \mathbf{A}_0 - \mathbf{A}\boldsymbol{\lambda}^k\right). \qquad (15)$$

In summary, the algorithm 1 highlights the step-by-step learning procedures of the SCR cascade. Fitting a SCR model simply consists of recursively: evaluate the local features at the current parameters estimate; compute the update (eq. 15) and iterate $\mathbf{r}^{k+1} = \mathbf{r}^k + \Delta\mathbf{r}^k$.

4. EXPERIMENTAL EVALUATION

The performance evaluation was conducted in several 'in the wild' databases. Such attribute means that the images were acquired in unconstrained scenarios, i.e. under variations of lighting, focus, facial expression, pose and occlusions. Four datasets were used in total: (1) The LFPW [37] database that has 811 (train) and 224 (test) images collected over web searches (68 landmarks [38]); (2) The HELEN [39] database holds 2000 (train) plus 330 (test) images taken from the flickr site (68 landmarks [38]); (3) The LFW [40], the largest set, has more than 13000 images (10 landmarks); The train/test portions had a split of 70/30; Finally, (4) the 300W [41] [38] consists of 300 images taken in both indoor and outdoor scenarios (with a combined test set of 600 images). The train set uses a combination of images from other datasets (AFW [24], HELEN, iBug [38], LFPW and XM2VTS [42]), making a total of 6197 images (68 landmarks).

The evaluation itself includes a comparison with some of the classical LK based techniques (briefly in section 2), a representative Constrained Local Model (CLM) method, the part-based Tree-Model (TM) [24] and some recent cascaded regression techniques. The classical/baseline techniques referred are the Simultaneous Forwards Additive (SFA), the Simultaneous Inverse Compositional (SIC), the Project-Out Inverse Compositional (PO-IC) and (for completeness) the Project-Out Forwards Additive (PO-FA). We would



(a) (b) (c) (d)

(%) Area under cdf curve / total area	LFPW	HELEN	LFW	300W
Initial Estimate [36]	46.4	41.6	61.7	27.2
PO-FA	53.6	51.3	67.3	38.2
SFA	70.0	60.2	73.0	42.3
PO-IC [11]	56.1	53.8	69.4	39.1
SIC [13]	73.1	63.5	75.6	43.9

(%) Area under cdf curve / total area	LFPW	HELEN	LFW	300W
SCMS [21] (grey level)	56.9	50.7	70.7	40.9
TM [24] (p146)	56.5	54.8	60.1	36.7
SDM [27]	72.2	69.7	81.5	50.3
PO-CR [28]	80.4	72.5	84.1	53.3
SCR (our method)	**82.6**	**74.8**	**85.5**	**55.5**

Fig. 2. Fitting performance curves in the (a) LFPW, (b) HELEN, (c) LFW and (d) 300W databases, respectively. The tables show a quantitative measure of the ratio between the area below each curve and the total area. The images on top, show fitting examples with our SCR technique.

like to remark that the warp function here involved is defined as in eq. 1, not the piece-wise affine warp commonly found in the AAM literature [2]. According, the PO-IC technique acts as a simplified version of Gauss-Newton Deformable Parts Models (GN-DPM) [25]. Regarding the CLM method, the Subspace Constrained Mean-Shift (SCMS) was used. Finally, the evaluation of the cascaded regression approaches includes the Supervised Descent Method (SDM) [27] and the Project-Out Cascaded Regression (PO-CR) [28] against our proposed Simultaneous Cascaded Regression (SCR).

All methods were trained with the same amount of data (independently for each database) and they were built with the same local appearance settings, namely HoG features [35] and a local support region size of 27×27 (cellsize = 3). A notable exception is the TM method which is based on the author's supplied model (p146). Under the same assumption, testing was made using the same initialization where the shape and pose parameters started with the mean shape and appearance, respectively and the pose parameters were obtained from a face detector [36]. Both LK based methods and SCMS were fitted until convergence up to a max of 30 iterations. The mentioned CLM (SCMS) has local landmark detectors that are based on MOSSE filters [43] built with grey level intensities. The response maps optimization include mean-shift updates with a kernel bandwidth schedule of $(15, 10, 5, 2)$.

Regarding the cascade regression methods, the number of cascade levels was established to be equal to $K = 5$. In SCR, the regularization parameters were set to $\lambda_1 = 10^{-3}$ and $\lambda_2 = 10^{-4}$, respectively. The density of the perturbations M (eq. 12), mainly for computation memory concerns, depends on the training dataset (as it requires to hold, in a matrix $\mathbf{E}$, all extracted feature data for all images, landmarks and parameter disturbances). In our experiments we were able to learn SDM, PO-CR and SCR models with $M = 20$ in LFPW, $M = 10$ in LFW and $M = 5$ in both HELEN and 300W.

As standard, the alignment error is measured by the mean error per landmark as fraction of the inter-ocular distance, d_{eyes}, as $e_m(\mathbf{s}) = \frac{1}{v\ d_{\text{eyes}}} \sum_{i=1}^{v} \|\mathbf{s}_i - \mathbf{s}_i^*\|$ where $\mathbf{s}_i^*$ is the location of i^{th} landmark in the ground truth annotation. Figure 2 shows the cumulative error distribution curves for all the evaluated methods and datasets. The table included in the figure shows a quantitative measure of the results which is defined as the ratio, in percentage, between the area bellow the fitting curve and the total area of a ground truth curve.

As expected, the results show that LK Simultaneous algorithms (SFA and SIC) perform better that the Project-Out versions (PO-FA and PO-IC). This performance advantage results from the enhanced of optimization strategy that scans for all parameters at once. The Inverse Compositional methods are slightly more robust mainly because of the gradients involved. In the last, the gradients are evaluated w.r.t. the appearance model (rather than the input image) which is less prone to noise. The TM was mostly designed to operate as detector. Its limited accuracy comes from the simple regularization used, which is made for fast inference (dynamic programming).

The results on cascaded regression techniques shows that: the SDM performs clearly better than any other non-regression method (as it captures the variance of the initialization), the PO-CR performed slightly better than SDM (because it acts as more constrained regression) and the SCR was able to outperform all the previous (due to the underlying shape and appearance structure included in the regression). Ultimately, like in its LK counterparts, we noted performance gains between the Simultaneous and the Project-Out version.

From the computational point of view, our approach takes an average of 115ms for running one cascade iteration step (MatLab).

5. CONCLUSIONS

This paper revisits the Lucas & Kanade image alignment formulation and introduces a cascaded regression extension of the Simultaneous algorithm. Our approach exploits the joint optimization structure and draws Newton based gradients as components of the regression. The results demonstrate the accuracy and effectiveness of our method.

6. REFERENCES

[1] T. F. Cootes, G. J. Edwards, and C. J. Taylor, "Active appearance models," *IEEE TPAMI*, vol. 23, no. 6, pp. 681–685, 2001.

[2] I. Matthews and S. Baker, "Active appearance models revisited," *IJCV*, vol. 60, no. 1, pp. 135–164, November 2004.

[3] G. Tzimiropoulos and M. Pantic, "Fast algorithms for fitting active appearance models to unconstrained images," *IJCV*, vol. 122, no. 1, pp. 17–33, 2017.

[4] J. Alabort-i-Medina and S. Zafeiriou, "A unified framework for compositional fitting of active appearance models," *IJCV*, vol. 121, no. 1, pp. 26–64, 2017.

[5] A. U. Batur and M. H. Hayes, "Adaptive active appearance models," *IEEE TIP*, vol. 14, no. 11, pp. 1707–1721, 2005.

[6] T. F. Cootes and C. J. Taylor, "An algorithm for tuning an active appearance model to new data," in *BMVC*, 2006.

[7] J. Saragih and R. Göcke, "Learning aam fitting through simulation," *PR*, vol. 42, no. 11, pp. 2628–2636, 2009.

[8] P. Tresadern, P. Sauer, and T. F. Cootes, "Additive update predictors in active appearance models," in *BMVC*, 2010.

[9] B. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision (darpa)," in *DARPA Image Understanding Workshop*, April 1981, pp. 121–130.

[10] S. Baker and I. Matthews, "Lucas-kanade 20 years on: A unifying framework," *IJCV*, vol. 56, no. 1, pp. 221–255, 2004.

[11] S. Baker, R. Gross, and I. Matthews, "Lucas kanade 20 years on: A unifying framework: Part 3," Tech. Rep. CMU-RI-TR-03-35, CMU Robotics Institute, November 2003.

[12] S. Baker and I. Matthews, "Equivalence and efficiency of image alignment algorithms," in *IEEE CVPR*, 2001.

[13] R. Gross, I. Matthews, and S. Baker, "Generic vs. person specific active appearance models," *IVC*, vol. 23, no. 12, pp. 1080–1093, November 2005.

[14] J. Gonzalez-Mora, N. Guil, E. L. Zapata, and F. De la Torre, "Efficient image alignment using linear appearance models," in *IEEE CVPR*, 2009.

[15] S. Lucey, R. Navarathna, A. B. Ashraf, and S. Sridharan, "Fourier lucas-kanade algorithm," *IEEE TPAMI*, vol. 35, no. 6, pp. 1383–1396, 2013.

[16] J. Xiao, S. Baker, I. Matthews, and T. Kanade, "Real-time combined 2d+3d active appearance models," in *IEEE CVPR*, 2004.

[17] P. Martins, R. Caseiro, and J. Batista, "Generative face alignment through 2.5d active appearance models," *CVIU*, vol. 117, no. 3, pp. 250–268, March 2013.

[18] E. Antonakos, J. Alabort-i-Medina, G. Tzimiropoulos, and S. Zafeiriou, "Feature-based lucas-kanade and active appearance models," *IEEE TIP*, vol. 24, no. 9, pp. 2617–2632, 2015.

[19] D. Cristinacce and T. F. Cootes, "Automatic feature localisation with constrained local models," *PR*, vol. 41, no. 10, pp. 3054–3067, 2008.

[20] Y. Wang, S. Lucey, and J. Cohn, "Enforcing convexity for improved alignment with constrained local models," in *IEEE CVPR*, 2008.

[21] J. Saragih, S. Lucey, and J. Cohn, "Deformable model fitting by regularized landmark mean-shifts," *IJCV*, vol. 91, no. 2, pp. 200–215, 2010.

[22] P. Martins, R. Caseiro, and J. Batista, "Non-parametric bayesian constrained local models," in *IEEE CVPR*, 2014.

[23] P. Martins, J. F. Henriques, R. Caseiro, and J. Batista, "Bayesian constrained local models revisited," *IEEE TPAMI*, vol. 38, no. 4, pp. 704–716, April 2016.

[24] X. Zhu and D. Ramanan, "Face detection, pose estimation, and landmark localization in the wild," in *IEEE CVPR*, 2012.

[25] G. Tzimiropoulos and M. Pantic, "Gauss-newton deformable part models for face alignment in-the-wild," in *IEEE CVPR*, 2014.

[26] X. Cao, Y. Wei, F. Wen, and J. Sun, "Face alignment by explicit shape regression," in *IEEE CVPR*, 2012.

[27] X. Xiong and F. De la Torre, "Supervised descent method and its application to face alignment," in *IEEE CVPR*, 2013.

[28] G. Tzimiropoulos, "Project-out cascaded regression with an application to face alignment," in *IEEE CVPR*, 2015.

[29] X. P. Burgos-Artizzu, P. Perona, and P. Dollár, "Robust face landmark estimation under occlusion," in *IEEE ICCV*, 2013.

[30] V. Kazemi and J. Sullivan, "One millisecond face alignment with an ensemble of regression trees," in *IEEE CVPR*, 2014.

[31] D. Lee, H. Park, and C. D. Yoo, "Face alignment using cascade gaussian process regression trees," in *IEEE CVPR*, 2015.

[32] A. Jourabloo and X. Liu, "Pose-invariant 3d face alignment," in *IEEE International Conference on Computer Vision*, 2015.

[33] S. Zhu, C. Li, C. Loy, and X. Tang, "Face alignment by coarse-to-fine shape searching," in *IEEE CVPR*, 2015.

[34] S. Ren, X. Cao, Y. Wei, and J. Sun, "Face alignment at 3000 fps via regressing local binary features," in *IEEE CVPR*, 2014.

[35] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *IEEE CVPR*, 2005.

[36] P. Viola and M. Jones, "Robust real-time object detection," *IJCV*, vol. 57, no. 2, pp. 137–154, July 2002.

[37] P. N. Belhumeur, D. W. Jacobs, D. J. Kriegman, and N. Kumar, "Localizing parts of faces using a consensus of exemplars," in *IEEE CVPR*, 2011.

[38] C. Sagonas, G. Tzimiropoulos, S. Zafeiriou, and M. Pantic, "300 faces in-the-wild challenge: The first facial landmark localization challenge," in *IEEE ICCV Workshop*, 2013.

[39] V. Le, J. Brandt, Z. Lin, L. Boudev, and T. S. Huang, "Interactive facial feature localization," in *ECCV*, 2012.

[40] G. B. Huang, M. Ramesh, T. Berg, and E. L. Miller, "Labeled faces in the wild: A database for studying face recognition in unconstrained environments," Tech. Rep. 07-49, University of Massachusetts, Amherst, 2007.

[41] C. Sagonas, E. Antonakos, G. Tzimiropoulos, and M. Pantic, "300 faces in-the-wild challenge: database and results," *IVC, Special Issue on Facial Landmark Localisation 'In-The-Wild'*, vol. 47, pp. 3–18, 2016.

[42] K. Messer, J. Matas, J. Kittler, J. Luettin, and G. Maitre, "XM2VTSDB: The extended M2VTS database," in *AVBPA*, 1999.

[43] D. S. Bolme, J. R. Beveridge, B. A. Draper, and Y. M. Lui, "Visual object tracking using adaptive correlation filters," in *IEEE CVPR*, 2010.