Robust and Efficient Parametric Face Alignment

Georgios Tzimiropoulos [†] Stefanos Zafeiriou[†] Maja Pantic [†,*]

[†]Dept. of Computing,
Imperial College London
180 Queen's Gate
London SW7 2AZ, U.K.
{gt204,s.zafeiriou,m.pantic}@imperial.ac.uk

[*]EEMCS
University of Twente
Drienerlolaan 5
7522 NB Enschede
The Netherlands [*]

Abstract

We propose a correlation-based approach to parametric object alignment particularly suitable for face analysis applications which require efficiency and robustness against occlusions and illumination changes. Our algorithm registers two images by iteratively maximizing their correlation coefficient using gradient ascent. We compute this correlation coefficient from complex gradients which capture the orientation of image structures rather than pixel intensities. The maximization of this gradient correlation coefficient results in an algorithm which is as computationally efficient as ℓ_2 norm-based algorithms, can be extended within the inverse compositional framework (without the need for Hessian re-computation) and is robust to outliers. To the best of our knowledge, no other algorithm has been proposed so far having all three features. We show the robustness of our algorithm for the problem of face alignment in the presence of occlusions and non-uniform illumination changes. The code that reproduces the results of our paper can be found at http://ibug.doc.ic.ac.uk/resources.

1. Introduction

Object alignment methods aim at finding the transformation or deformation which minimizes the discrepancies between two or more images/objects. In automated face analysis, these discrepancies usually stem from rigid head motions induced by observing faces at different time instances and from different viewpoints as well as from non-rigid facial deformations induced by facial expressions. Alignment methods aim at estimating these motions and, therefore, play a central role in the efficacy and robustness of high-level applications such as face recognition, speech reading and facial expression analysis.

In this work, we focus on object alignment methods based on gradient descent optimization. Since the first algorithm of this type, the Lucas-Kanade (LK) algorithm [1], gradient descent has become one of the key ingredients in face alignment algorithms. Numerous extensions to the LK algorithm have been proposed to address issues related to efficiency [2–5], generalization capacity [2, 6, 7], optimization [8, 9] and robustness [10–13]. Most prior work is based on ℓ_2 norm minimization. The ℓ_2 norm is the standard choice [1, 2, 5, 13, 14], as it can result in computationally efficient algorithms. Perhaps, the most notable example of such algorithms is the inverse compositional algorithm proposed by Baker and Matthews [4, 5]. At each iteration, the method solves a linear least squares problem with the Hessian pre-computed and constant across iterations. As usual, the choice of the norm imposes a trade-off between robustness and computational complexity. Robust approaches typically replace the ℓ_2 norm with a robust error function [10, 11]. Such methods solve a re-weighted least squares problem, where the weights are updated at each iteration. This additional computation makes them much slower. For example, replacing the ℓ_2 norm with a robust function within the inverse compositional framework, requires the re-computation of the Hessian at each iteration, resulting in a less efficient algorithm [11].

In this paper, we propose a new cost function for gradient ascend face alignment: the maximization of the correlation of image gradient orientations. The use of this correlation coefficient has been motivated by the recent success of FFT-based gradient correlation methods for the robust estimation of translational displacements [15–17]. More specifically, we use a correlation coefficient which takes the form of the sum of cosines of gradient orientation differences. The use of gradient orientation differences is the key to the robustness of the proposed scheme. As it is was shown in [15,18], local orientation mismatches caused by outliers can be well-described by a uniform distribution which, under a number

[*]This work has been funded by the European Research Council under the ERC Starting Grant agreement no. ERC- 2007-StG-203143 (MAHNOB).

of mild assumptions, is canceled out by applying the cosine kernel. Thus, image regions corrupted by outliers result in approximately zero correlation and therefore do not bias the estimation of the transformation parameters significantly.

To maximize the gradient correlation coefficient, we formulate and solve a continuous optimization problem. The proposed methodology results in a computationally efficient and robust alignment algorithm. In particular, our algorithm is as efficient as ℓ_2 norm-based algorithms, can be extended within the inverse compositional framework (without the need for Hessian re-computation) and is robust to outliers caused by occlusions and non-uniform illumination changes. To the best of our knowledge, no other algorithm has been previously proposed having all three features.

To evaluate the performance of our scheme, we considered the problem of face alignment in the presence of occlusions and non-uniform illumination changes using hundreds of real face pairs taken from the AR [19] and Yale B [20] databases. Our results show that, unlike previously proposed schemes, our algorithm can cope with such cumbersome problems.

Summarizing our contributions, in this paper

- We propose the maximization of the correlation of image gradient orientations as a new cost function for robust gradient ascent face alignment.
- We formulate and solve the continuous optimization problems which result in the forward additive and inverse compositional versions of our algorithm.
- We present results for very challenging alignment cases which have not been previously examined. Table 1 presents a comparison between our experiments and the ones reported in related alignment papers. The code that reproduces the results of our paper can be found at http://ibug.doc.ic.ac.uk/resources.

2. Gradient-based correlation coefficient

Assume that we are given the image-based representations of two objects $\mathbf{I}_i \in \Re^{m_1 \times m_2}$, $i = 1, 2$. We define the complex representation which combines the magnitude and the orientation of image gradients as $\mathbf{G}_i = \mathbf{G}_{i,x} + j\mathbf{G}_{i,y}$, where $j = \sqrt{(-1)}$,$\mathbf{G}_{i,x} = \mathbf{F}_x \star \mathbf{I}_i$, $\mathbf{G}_{i,y} = \mathbf{F}_y \star \mathbf{I}_i$ and $\mathbf{F}_x, \mathbf{F}_y$ are filters used to approximate the ideal differentiation operator along the image horizontal and vertical direction respectively. We also denote by $\mathcal{P}$ the set of indices corresponding to the image support and by $\mathbf{g}_i = \mathbf{g}_{i,x} + j\mathbf{g}_{i,y}$ the $N-$dimensional vectors obtained by writing $\mathbf{G}_i$ in lexicographic ordering, where N is the cardinality of $\mathcal{P}$. The gradient correlation coefficient is defined as

$$s \triangleq \Re\{\mathbf{g}_1^H \mathbf{g}_2\}, \tag{1}$$

where $\Re\{.\}$ denotes the real part of a complex number and H denotes the conjugate transpose [15]. Using $\mathbf{r}_i(k) \triangleq \sqrt{\mathbf{g}_{i,x}^2(k) + \mathbf{g}_{i,y}^2(k)}$ and $\phi_i(k) \triangleq \arctan \frac{\mathbf{g}_{i,y}(k)}{\mathbf{g}_{i,x}(k)}$, we have

$$s \triangleq \sum_{k \in \mathcal{P}} \mathbf{r}_1(k)\mathbf{r}_2(k) \cos[\Delta\boldsymbol{\phi}(k)], \tag{2}$$

where $\Delta\boldsymbol{\phi} \triangleq \boldsymbol{\phi}_1 - \boldsymbol{\phi}_2$.

The magnitudes $\mathbf{r}_i$ in (2) suppress the contribution of areas of constant intensity level which do not provide useful features for object alignment. Note, however, that the use of gradient magnitude does not necessarily result in robust algorithms. For example, the authors in [21] have shown that the gradient magnitude varies drastically with the change in the direction of the light source.

The key to the robustness of the proposed scheme is the correlation of gradient orientations which takes the form of the sum of cosines of gradient orientation differences [15, 17]. To show this [15, 18], assume that there exists a subset $\mathcal{P}_o \subset \mathcal{P}$ corresponding to the set of pixels corrupted by outliers. By using the normalized gradients $\tilde{\mathbf{g}}_i = \tilde{\mathbf{g}}_{i,x} + j\tilde{\mathbf{g}}_{i,y}$, where $\tilde{\mathbf{g}}_{i,x}(k) = \mathbf{g}_{i,x}(k)/|\mathbf{g}_i(k)|$ and $\tilde{\mathbf{g}}_{i,y}(k) = \mathbf{g}_{i,y}(k)/|\mathbf{g}_i(k)|$, so that $\mathbf{r}_i(k) = 1\ \forall k$, the value of this gradient correlation coefficient in $\mathcal{P}_o$ is

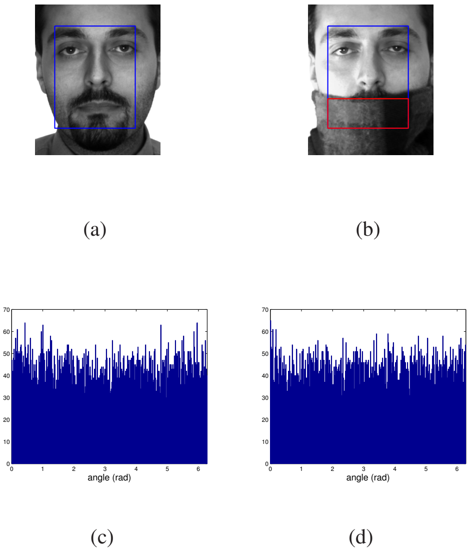
$$q_o \triangleq \sum_{k \in \mathcal{P}_o} \cos[\Delta\boldsymbol{\phi}(k)]. \tag{3}$$

To compute the value of q_o, we note that in $\mathcal{P}_o$ the images are *visually dissimilar/unrelated*, so that locally do not match. It is therefore not unreasonable to assume that for any spatial location k, the difference in gradient orientation $\Delta\boldsymbol{\phi}(k)$ can take any value in the range $[0, 2\pi)$ with equal probability. Thus, we can assume that $\Delta\boldsymbol{\phi}$ is a realization of a stationary random process $u(t)$ which $\forall t$ follows a uniform distribution $U(0, 2\pi)$. Given this, it is not difficult to show that, under some rather mild assumptions, it holds

$$q_o = \sum_{k \in \mathcal{P}_o} \cos[\Delta\boldsymbol{\phi}(k)] \simeq 0. \tag{4}$$

This assumption has been shown to be valid using the Kolmogorov-Smirnoff test for more than 70.000 pairs of *visually unrelated images* in [18]. As an example, in Fig. 1 (a)-(b), we assume that the scarf is visually unrelated to the face. $\mathcal{P}_o$ here corresponds to the part of the face occluded by the scarf defined by the red rectangle. Fig. 1 (c) plots the distribution of $\Delta\boldsymbol{\phi}$ in $\mathcal{P}_o$, while Fig. 1 (d) shows the histogram of uniformly distributed samples obtained with Matlab's rand function. As in [18], to verify that $\Delta\boldsymbol{\phi}$ is uniformly distributed, we used the Kolmogorov-Smirnov test [22] to test the null hypothesis $H_0 : \forall k \in \mathcal{P}_o,\ \Delta\boldsymbol{\phi}(k) \sim U[0, 2\pi)$. For a significance level of 0.01, the null hypothesis was accepted with p-value equal to 0.254. Similarly, for the samples obtained with Matlab's `rand` function, the null hypothesis was accepted with $p = 0.48$.

Overall, unlike standard correlation (i.e. the inner product) of pixel intensities where the contribution of outliers can be arbitrarily large, the effect of outliers is approximately canceled out in $\mathcal{P}_o$. Corrupted regions result in approximately zero correlation and thus do not bias the estimation of the transformation parameters.


(a)

(b)

(c)

(d)

Figure 1. (a)-(b) A pair of faces from the AR database. The region of interest is defined by the blue rectangle. The corrupted region $\mathcal{P}_o$ is defined by the red rectangle. (c) The distribution of $\Delta\phi$ in $\mathcal{P}_o$. (d) The distribution of samples (uniformly distributed) obtained with Matlab's rand function.

3. Gradient Orientation in Face Analysis

The use of gradient orientation as useful features for face analysis is by no means proposed for the first time in this work. Examples of previous work can be found in [21, 23, 24]. However, most prior work proposes gradient orientations as features for achieving insensitivity in non-uniform illumination variations. On the contrary, what is highlighted in [15, 18] as well as in this work is why gradient orientations can be used for outlier-robust (for example occlusion-robust) face analysis.

Regarding face alignment, perhaps what is somewhat related to the proposed scheme is the Active Appearance Model proposed in [24]. We underline two important differences between our algorithm and the method of [24]. First, as [24] *does* employ the gradient magnitude (even for normalization) for feature extraction, it is inevitably less robust to outliers. Second, no attempt to exploit the relation between image gradients and pixel intensities is made. More specifically, the gradient-based features in [24] are treated just as pixel intensities which are then used for *regression-based* object alignment. On the contrary, we make full use of the relation between image gradients and pixel intensities to formulate and solve a *continuous optimization* problem. This results in a dramatic performance improvement as Section 5 illustrates.

4. Robust and efficient object alignment

Parametric object alignment methods assume that $\mathbf{I}_1$ and $\mathbf{I}_2$ are related by a parametric transformation, i.e.

$$\mathbf{I}_1(\mathbf{x}_k) = \mathbf{I}_2(\mathbf{W}(\mathbf{x}_k; \mathbf{p})), \forall k \in \mathcal{P}, \tag{5}$$

where $\mathbf{W}(\mathbf{x}_k; \mathbf{p})$ is the parametric transformation with respect to the image coordinates $\mathbf{x}_k = [\mathbf{x}_1(k), \mathbf{x}_2(k)]^T$ and $\mathbf{p} = [\mathbf{p}(1), \ldots, \mathbf{p}(n)]^T$ is the vector of the unknown parameters. Next, $\mathbf{p}$ is estimated by minimizing an objective function which is typically the ℓ_2 norm of the difference $\mathbf{E} = \mathbf{I}_1 - \mathbf{I}_2$. The minimization is performed in an iterative fashion after making a first or second order Taylor approximation to either $\mathbf{I}_1$ or $\mathbf{I}_2$.

4.1. The quantity maximized

In this section, we propose the maximization of the correlation of image gradient orientations as a new cost function for robust gradient descent face alignment. In particular, to estimate $\mathbf{p}$, we wish to maximize

$$q = \sum_{k \in \mathcal{P}} \cos[\Delta\phi(k)]. \tag{6}$$

By using the normalized gradients $\tilde{\mathbf{g}}_i$, simple calculations show that (6) is equivalent to

$$q = \sum_{k \in \mathcal{P}} \tilde{\mathbf{g}}_{1,x}(k)\tilde{\mathbf{g}}_{2,x}(k) + \tilde{\mathbf{g}}_{1,y}(k)\tilde{\mathbf{g}}_{2,y}(k). \tag{7}$$

Note, however, that a first order Taylor expansion of $\tilde{\mathbf{g}}_1$ or $\tilde{\mathbf{g}}_2$ with respect to $\Delta\mathbf{p}$ yields a linear function of $\Delta\mathbf{p}$ which is maximized as $\Delta\mathbf{p} \to \infty$. To alleviate this problem without resorting to the second order Taylor expansion as in [25], we follow an approach similar to [14]. To proceed, we note that as $||\tilde{\mathbf{g}}_2(k)||_2 = 1, \forall k \in \mathcal{P}$, the proposed cost function is exactly equal to

$$q = \frac{\sum_{k \in \mathcal{P}} \tilde{\mathbf{g}}_{1,x}(k)\tilde{\mathbf{g}}_{2,x}(k) + \tilde{\mathbf{g}}_{1,y}(k)\tilde{\mathbf{g}}_{2,y}(k)}{\sqrt{\sum_{k \in \mathcal{P}} \tilde{\mathbf{g}}_{2,x}^2(k) + \tilde{\mathbf{g}}_{2,y}^2(k)}}, \tag{8}$$

but if we linearize $\tilde{\mathbf{g}}_2$ in the above expression, the denominator will not be equal to 1 and q will become a non-linear function of $\Delta\mathbf{p}$. Finally, using vector notation, our cost function becomes

$$q = \frac{\tilde{\mathbf{g}}_{1,x}^T\tilde{\mathbf{g}}_{2,x} + \tilde{\mathbf{g}}_{1,y}^T\tilde{\mathbf{g}}_{2,y}}{\sqrt{\tilde{\mathbf{g}}_{2,x}^T\tilde{\mathbf{g}}_{2,x} + \tilde{\mathbf{g}}_{2,y}^T\tilde{\mathbf{g}}_{2,y}}}. \tag{9}$$

To maximize q with respect to $\mathbf{p}$, we first make the dependence of $\tilde{\mathbf{g}}_2(k)$ on $\mathbf{p}$ explicit by writing $\tilde{\mathbf{g}}_2[\mathbf{p}](k)$. Then, we maximize iteratively by assuming that the current estimate of $\mathbf{p}$ is known and by looking for an increment $\Delta\mathbf{p}$ which maximizes our objective function in (9) with respect to $\Delta\mathbf{p}$.

4.2. The forward-additive gradient correlation algorithm

In this section, we describe how to maximize the proposed cost function in (9) using the forward-additive maximization procedure. In this framework [1, 5], at each iteration, we maximize (9) with respect to $\Delta\mathbf{p}$ where $\mathbf{g}_2 \longleftarrow \mathbf{g}_2[\mathbf{p}+\Delta\mathbf{p}]$. Once we obtain $\Delta\mathbf{p}$, we update the parameter vector in an additive fashion $\mathbf{p} \longleftarrow \mathbf{p}+\Delta\mathbf{p}$ and use this new value of $\mathbf{p}$ to obtain the updated warped image $\mathbf{I}_2(\mathbf{W}(\mathbf{x};\mathbf{p}))$.

We start by noting that $\mathbf{g}_2[\mathbf{p}](k)$ is the complex gradient of $\mathbf{I}_2(\mathbf{W}(\mathbf{x};\mathbf{p}))$ with respect to the original coordinate system evaluated at $\mathbf{x} = \mathbf{x}_k$. This gradient is different from the gradient of $\mathbf{I}_2$ calculated at the first iteration and then evaluated at $\mathbf{W}(\mathbf{x}_k;\mathbf{p})$, which, for convenience, we will denote by $\mathbf{h}_2[\mathbf{p}](k)$. That is, $\mathbf{h}_2[\mathbf{p}] = \mathbf{h}_{2,x}[\mathbf{p}] + j\mathbf{h}_{2,y}[\mathbf{p}]$ is obtained by writing $\mathbf{G}_{2,x}(\mathbf{W}(\mathbf{x};\mathbf{p})) + j\mathbf{G}_{2,y}(\mathbf{W}(\mathbf{x};\mathbf{p}))$ in lexicographic ordering, where $\mathbf{G}_2 = \mathbf{G}_{2,x} + j\mathbf{G}_{2,y}$ is assumed to be computed at the first iteration. In a similar fashion, we denote by $\mathbf{h}_{2,xx}[\mathbf{p}]$, $\mathbf{h}_{2,yy}[\mathbf{p}]$ and $\mathbf{h}_{2,xy}[\mathbf{p}]$, the vectors obtained by writing in lexicographic ordering the second partial derivatives of I_2, $\mathbf{G}_{2,xx}$, $\mathbf{G}_{2,yy}$ and $\mathbf{G}_{2,xy}$, computed at the first iteration and, then, evaluated at $\mathbf{W}(\mathbf{x};\mathbf{p})$. Let us also write $\mathbf{W}(\mathbf{x};\mathbf{p}) = [\mathbf{w}_1(\mathbf{x};\mathbf{p}), \mathbf{w}_2(\mathbf{x};\mathbf{p})]^T$, so that the matrix derivative with respect to a vector $\mathbf{a} = [\mathbf{a}(1), \ldots, \mathbf{a}(m)]^T$ is given by

$$\frac{\partial\mathbf{W}}{\partial\mathbf{a}} = \begin{bmatrix} \frac{\partial\mathbf{w}_1}{\partial\mathbf{a}(1)} & \cdots & \frac{\partial\mathbf{w}_1}{\partial\mathbf{a}(m)} \\ \frac{\partial\mathbf{w}_2}{\partial\mathbf{a}(1)} & \cdots & \frac{\partial\mathbf{w}_2}{\partial\mathbf{a}(m)} \end{bmatrix}. \tag{10}$$

By definition we have

$$\begin{aligned} \mathbf{g}_2[\mathbf{p}](k) &\triangleq [\mathbf{g}_{2,x}[\mathbf{p}](k) \;\; \mathbf{g}_{2,y}[\mathbf{p}](k)] \\ &\triangleq \left.\frac{\partial\mathbf{I}_2(\mathbf{W}(\mathbf{x};\mathbf{p}))}{\partial\mathbf{x}}\right|_{\mathbf{x}=\mathbf{x}_k} \\ &= \nabla_{\mathbf{W}}\mathbf{I}_2[\mathbf{p}](k) \left.\frac{\partial\mathbf{W}}{\partial\mathbf{x}}\right|_{\mathbf{x}=\mathbf{x}_k}, \end{aligned} \tag{11}$$

where $\nabla_{\mathbf{W}}\mathbf{I}_2[\mathbf{p}](k) \triangleq [\mathbf{h}_{2,x}[\mathbf{p}](k) \;\; \mathbf{h}_{2,y}[\mathbf{p}](k)]$. By applying the chain rule and noticing that $\nabla_{\mathbf{W}}\frac{\partial\mathbf{W}}{\partial\mathbf{x}} = 0$, we also have

$$\begin{aligned} \begin{bmatrix} \frac{\partial\mathbf{g}_{2,x}[\mathbf{p}](k)}{\partial\mathbf{p}} \\ \frac{\partial\mathbf{g}_{2,y}[\mathbf{p}](k)}{\partial\mathbf{p}} \end{bmatrix} &= \left(\left.\frac{\partial\mathbf{W}}{\partial\mathbf{x}}\right|_{\mathbf{x}=\mathbf{x}_k} \right)^T \\ &\times \begin{bmatrix} \mathbf{h}_{2,xx}[\mathbf{p}](k) & \mathbf{h}_{2,xy}[\mathbf{p}](k) \\ \mathbf{h}_{2,yx}[\mathbf{p}](k) & \mathbf{h}_{2,yy}[\mathbf{p}](k) \end{bmatrix} \frac{\partial\mathbf{W}}{\partial\mathbf{p}}. \end{aligned} \tag{12}$$

We assume that the current estimate of $\mathbf{p}$ is known. The key point to make derivations tractable is to recall that $\tilde{\mathbf{g}}_{2,x}[\mathbf{p}](k) \equiv \cos\boldsymbol{\phi}_2[\mathbf{p}](k)$ and $\tilde{\mathbf{g}}_{2,y}[\mathbf{p}](k) \equiv \sin\boldsymbol{\phi}_2[\mathbf{p}](k)$ where

$$\boldsymbol{\phi}_2[\mathbf{p}](k) = \arctan\frac{\mathbf{g}_{2,y}[\mathbf{p}](k)}{\mathbf{g}_{2,x}[\mathbf{p}](k)}. \tag{13}$$

By performing a first order Taylor expansion on $\tilde{\mathbf{g}}_{2,x}[\mathbf{p}+\Delta\mathbf{p}](k)$, we get

$$\tilde{\mathbf{g}}_{2,x}[\mathbf{p}+\Delta\mathbf{p}](k) \approx \cos\boldsymbol{\phi}_2[\mathbf{p}](k) + \frac{\partial\cos\boldsymbol{\phi}_2[\mathbf{p}](k)}{\partial\mathbf{p}}\Delta\mathbf{p}. \tag{14}$$

By repeatedly applying the chain rule, we get

$$\frac{\partial\cos\boldsymbol{\phi}_2[\mathbf{p}](k)}{\partial\mathbf{p}} = -\sin\boldsymbol{\phi}_2[\mathbf{p}](k)\mathbf{j}[\mathbf{p}](k), \tag{15}$$

where $\mathbf{j}[\mathbf{p}](k)$ is a $1 \times n$ vector given by

$$\mathbf{j}[\mathbf{p}](k) = \frac{\cos\boldsymbol{\phi}_2[\mathbf{p}](k)\frac{\partial\mathbf{g}_{2,y}[\mathbf{p}](k)}{\partial\mathbf{p}} - \sin\boldsymbol{\phi}_2[\mathbf{p}](k)\frac{\partial\mathbf{g}_{2,x}[\mathbf{p}](k)}{\partial\mathbf{p}}}{\sqrt{\mathbf{g}_{2,x}^2[\mathbf{p}](k) + \mathbf{g}_{2,y}^2[\mathbf{p}](k)}}. \tag{16}$$

Using vector notation, we can write

$$\tilde{\mathbf{g}}_{2,x}[\mathbf{p}+\Delta\mathbf{p}] \approx \cos\boldsymbol{\phi}_2[\mathbf{p}] - \mathbf{S}_\phi[\mathbf{p}] \odot \mathbf{J}[\mathbf{p}]\Delta\mathbf{p}, \tag{17}$$

where $\mathbf{S}_\phi[\mathbf{p}]$ is the $N \times n$ matrix whose $k-$th row has n elements all equal to $\sin\boldsymbol{\phi}_2[\mathbf{p}](k)$, $\mathbf{J}[\mathbf{p}]$ is the $N \times n$ Jacobian matrix whose $k-$th row has n elements corresponding to $\mathbf{j}[\mathbf{p}](k)$ and $\odot$ denotes the Hadamard product. Very similarly, we can derive

$$\tilde{\mathbf{g}}_{2,y}[\mathbf{p}+\Delta\mathbf{p}] \approx \sin\boldsymbol{\phi}_2[\mathbf{p}] + \mathbf{C}_\phi[\mathbf{p}] \odot \mathbf{J}[\mathbf{p}]\Delta\mathbf{p}, \tag{18}$$

where $\mathbf{C}_\phi[\mathbf{p}]$ is the $N \times n$ matrix whose $k-$th row has n elements all equal to $\cos\boldsymbol{\phi}_2[\mathbf{p}](k)$.

Let us denote by $\mathbf{S}_{\Delta\phi}[\mathbf{p}]$ the $N \times 1$ vector whose $k-$th element is equal to $\sin(\boldsymbol{\phi}_1(k) - \boldsymbol{\phi}_2[\mathbf{p}](k))$. Then, by plugging (17) and (18) into (9), and after some calculations, our cost function becomes

$$q(\Delta\mathbf{p}) = \frac{q_{\mathbf{p}} + \mathbf{S}_{\Delta\phi}^T\mathbf{J}\Delta\mathbf{p}}{\sqrt{N + \Delta\mathbf{p}^T\mathbf{J}^T\mathbf{J}\Delta\mathbf{p}}}, \tag{19}$$

where $q_{\mathbf{p}} = \cos\boldsymbol{\phi}_1^T\cos\boldsymbol{\phi}_2 + \sin\boldsymbol{\phi}_1^T\sin\boldsymbol{\phi}_2$ is the correlation of gradient orientations between $\mathbf{I}_1$ and $\mathbf{I}_2(\mathbf{W}(\mathbf{x};\mathbf{p}))$, and we have dropped the dependence of the quantities on $\mathbf{p}$ for notational simplicity. Finally, the maximization of (19) with respect to $\Delta\mathbf{p}$ can be obtained by applying the results of [14]. In particular, the maximum value is attained for

$$\Delta\mathbf{p} = \lambda(\mathbf{J}^T\mathbf{J})^{-1}\mathbf{J}^T\mathbf{S}_{\Delta\phi}, \tag{20}$$

where $\lambda = \frac{1}{\tilde{q}}$ and $\tilde{q} = q_{\mathbf{p}}/N$ denotes the normalized correlation (such that $|\tilde{q}| \leq 1$) Thus, λ has a very intuitive interpretation. As $\tilde{q}$ is small (large) in the first (last) iterations, a large (small) λ is used as a weight in (20).

4.3. The inverse-compositional gradient correlation algorithm

In this section, we show how to maximize the proposed cost function in (9) using the inverse-compositional maximization procedure. In this framework [4, 5], a change of variables is made to switch the roles of $\mathbf{I}_1$ and $\mathbf{I}_2$ and the updated warp is obtained in a compositional (rather than additive) fashion. Thus, our cost function becomes

$$q = \frac{(\tilde{\mathbf{g}}_{2,x}[\mathbf{p}])^T(\tilde{\mathbf{g}}_{1,x}[\Delta\mathbf{p}]) + (\tilde{\mathbf{g}}_{2,y}[\mathbf{p}])^T(\tilde{\mathbf{g}}_{1,y}[\Delta\mathbf{p}])}{\sqrt{(\tilde{\mathbf{g}}_{1,x}[\Delta\mathbf{p}])^T(\tilde{\mathbf{g}}_{1,x}[\Delta\mathbf{p}]) + (\tilde{\mathbf{g}}_{1,y}[\Delta\mathbf{p}])^T(\tilde{\mathbf{g}}_{1,y}[\Delta\mathbf{p}])}} \tag{21}$$

with respect to $\Delta\mathbf{p}$ and, at each iteration, $\mathbf{I}_2$ is updated using $\mathbf{W}(\mathbf{x};\mathbf{p}) \longleftarrow \mathbf{W}(\mathbf{x};\mathbf{p}) \circ (\mathbf{W}(\mathbf{x};\Delta\mathbf{p}))^{-1}$, where $\circ$ denotes composition.

Similarly to [5], we assume that $\mathbf{W}(\mathbf{x};\mathbf{0}) = \mathbf{x}$. This, in turn, implies $\mathbf{g}_1[\Delta\mathbf{p}] \equiv \mathbf{h}_1[\Delta\mathbf{p}]$ which greatly simplifies the derivations. As before, we perform a Taylor approximation to $\tilde{\mathbf{g}}_{1,x}[\mathbf{p}]$, but this time around zero. This gives

$$\tilde{\mathbf{g}}_{1,x}[\Delta\mathbf{p}] \approx \cos\boldsymbol{\phi}_1[\mathbf{0}] - \mathbf{S}_\phi[\mathbf{0}] \odot \mathbf{J}[\mathbf{0}]\Delta\mathbf{p}, \tag{22}$$

where $\mathbf{S}_\phi[\mathbf{0}]$ is the $N \times n$ matrix whose $k-$th row has n elements all equal to $\sin\boldsymbol{\phi}_1[\mathbf{0}](k)$ and $\mathbf{J}[\mathbf{0}]$ is the $N \times n$ Jacobian matrix whose $k-$th row has n elements corresponding to the $1 \times n$ vector

$$\mathbf{j}[\mathbf{0}](k) = \frac{\cos\boldsymbol{\phi}_1[\mathbf{0}](k)\frac{\partial\mathbf{g}_{1,y}[\mathbf{0}](k)}{\partial\mathbf{p}} - \sin\boldsymbol{\phi}_1[\mathbf{0}](k)\frac{\partial\mathbf{g}_{1,x}[\mathbf{0}](k)}{\partial\mathbf{p}}}{\sqrt{\mathbf{g}^2_{1,x}[\mathbf{0}](k) + \mathbf{g}^2_{1,y}[\mathbf{0}](k)}} \tag{23}$$

and

$$\begin{bmatrix} \frac{\partial\mathbf{g}_{1,x}[\mathbf{0}](k)}{\partial\mathbf{p}} \\ \frac{\partial\mathbf{g}_{1,y}[\mathbf{0}](k)}{\partial\mathbf{p}} \end{bmatrix} = \begin{bmatrix} \mathbf{g}_{1,xx}[\mathbf{0}](k) & \mathbf{g}_{1,xy}[\mathbf{0}](k) \\ \mathbf{g}_{1,yx}[\mathbf{0}](k) & \mathbf{g}_{1,yy}[\mathbf{0}](k) \end{bmatrix} \left.\frac{\partial\mathbf{W}}{\partial\mathbf{p}}\right|_{\mathbf{p}=\mathbf{0}}.$$

Similarly, for $\tilde{\mathbf{g}}_{1,y}[\Delta\mathbf{p}]$, we get

$$\tilde{\mathbf{g}}_{1,y}[\Delta\mathbf{p}] \approx \sin\boldsymbol{\phi}_1[\mathbf{0}] + \mathbf{C}_\phi[\mathbf{0}] \odot \mathbf{J}[\mathbf{0}]\Delta\mathbf{p}, \tag{24}$$

where $\mathbf{C}_\phi[\mathbf{0}]$ is the $N \times n$ matrix whose $k-$th row has n elements all equal to $\cos\boldsymbol{\phi}_1[\mathbf{0}](k)$. Notice that all terms in (22) and (24) do not depend on $\mathbf{p}$ and, thus, are pre-computed and constant across iterations.

Let us denote by $\mathbf{S}_{\Delta\phi}[\mathbf{p}]$ the $N \times 1$ vector whose $k-$th element is equal to $\sin(\boldsymbol{\phi}_2[\mathbf{p}](k) - \boldsymbol{\phi}_1(k))$. Then, by dropping the dependence of the above quantities on $\mathbf{p}$ and $\mathbf{0}$, our objective function will be again given by (19) while the optimum $\Delta\mathbf{p}$ will be given by (20).

4.4. Computational complexity

A simple inspection of our algorithms shows that the most computationally expensive step is the calculation of $\mathbf{J}^T\mathbf{J}$ in (19) which requires $O(n^2N)$ operations. The cost of all other steps is at most $O(nN)$ (since $N \gg n$). In the inverse compositional maximization procedure, $\mathbf{J}^T\mathbf{J}$ and its inverse is pre-computed and, therefore, the complexity per iteration is $O(nN)$. Finally, an un-optimized MATLAB version of our algorithm takes about 0.03-0.04 seconds per iteration while the original inverse compositional algorithm takes about 0.02-0.03 seconds per iteration. We note that an optimized version of the original inverse compositional algorithm, as the core part of Active Appearance Model fitting, has been shown to track faces faster than 200 fps [7].

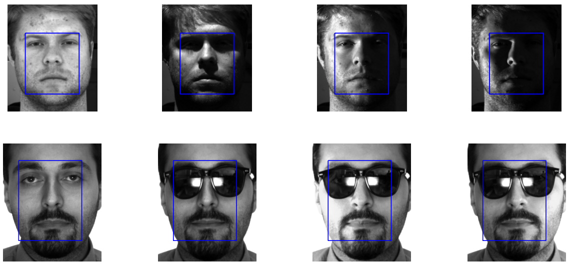


Figure 2. Examples of images used in our experiments (prior to the application of an affine transformation). The blue rectangle defines the region of interest.

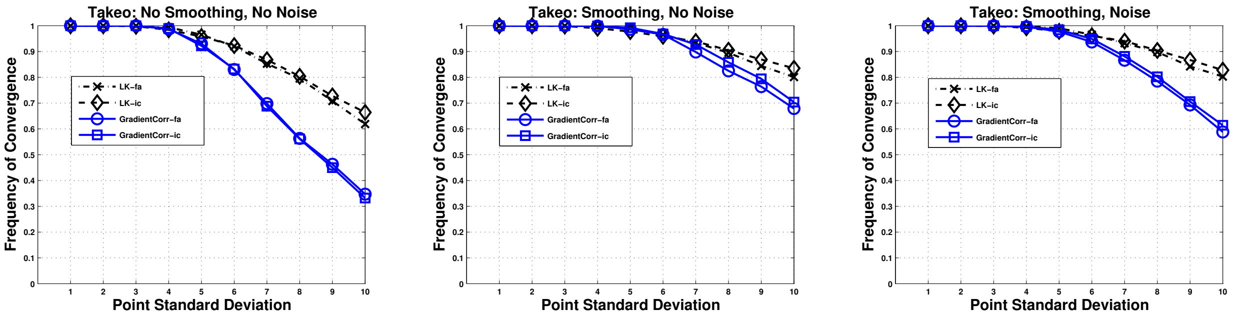
5. Face alignment experiments

We assessed the performance of our algorithms, which we coin GradientCorr-FA and GradientCorr-IC, using the performance evaluation framework proposed in [5] which has now become the standard evaluation procedure [9, 12–14]. We present results and comparison with previous work for very challenging alignment cases which have not been previously examined. Table 1 presents a comparison between our experiments and the ones reported in object alignment papers which also adopt the evaluation framework of [5]. In addition to the standard “Takeo” experiment, we considered, for the first time (to the best of our knowledge), the problem of face alignment in the presence of real occlusions and non-uniform illumination changes using hundreds of real faces taken from the AR [19] and Yale B [20] databases.

The evaluation in [5] is as follows. We selected a region of interest and three canonical points in this region. We perturbed these points using Gaussian noise of standard deviation σ and computed the initial RMS error between the canonical and perturbed points. Using the affine warp that the original and perturbed points defined, we generated the affine distorted image. Given a warp estimate, we computed the destination of the three canonical points and, then, the final RMS error between the estimated and correct locations. We used the average rate of convergence for a fixed σ and the average frequency of convergence for $\sigma = [1, 10]$ as the performance evaluation measures. An algorithm was con-

Methods	Number of image pairs considered	Real image pair	Transformation Affine/ Homography	Illumination	Occlusion	AWGN	Compared with
[5]	4 (Takeo+3)	No	Yes/Yes	No	No	Yes	[5]
[11]	6 (Takeo)	No	Yes/No	No	Yes (synthetic)	No	[5, 11]
[12]	3	Yes	Yes/No	Yes (natural)	No	No	[5]
[14]	1 (Takeo)	No	Yes/No	Yes (synthetic)	No	Yes	[5, 6]
[13]	NA (Multi-Pie [26])	Yes	Yes/No	Yes (natural)	No	No	[5]
[9]	11	No	Yes/No	Yes (synthetic)	No	Yes	[5]
Proposed	182 (Takeo + Yale +AR)	Yes	Yes/No	Yes (natural)	Yes (real)	Yes	[5, 11, 13, 14]

Table 1. Comparison between the experimental settings reported in object alignment papers following the evaluation framework of [5].

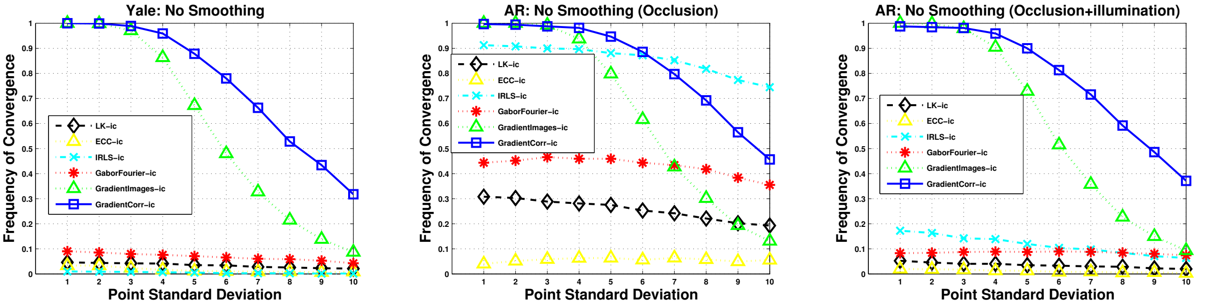


(a)

(b)

(c)

Figure 3. Frequency of Convergence vs Point Standard Deviation for Takeo image [5]. (a) No Smoothing, No Noise (b) Smoothing, No Noise (c) Smoothing, Noise. LK-fa: black-x. LK-IC: black-◇ . GradientCorr-fa: blue-○. GradientCorr-IC: blue-□.



(a)

(b)

(c)

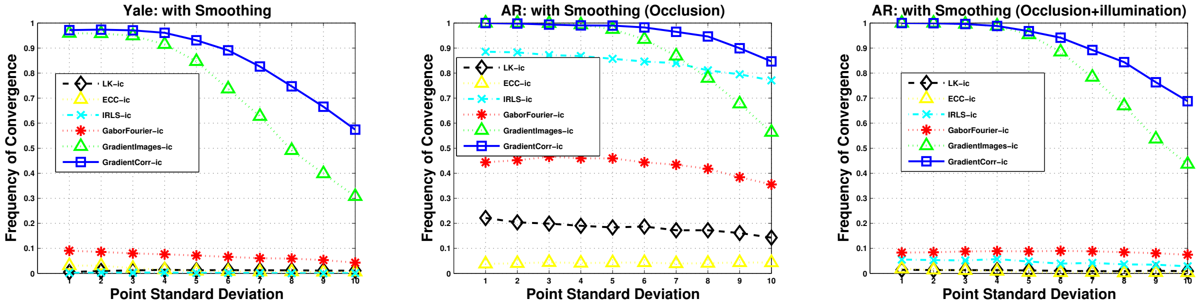
Figure 4. Average Frequency of Convergence vs Point Standard Deviation for Yale and AR databases. **No smoothing was used**. (a) Yale (b) AR-Occlusion (c) AR-Occlusion+illumination. LK-IC: black-◇. ECC-IC: yellow-△. IRLS-IC: cyan-x. GaborFourier-IC: red-*. GradientImages-IC: green-△. GradientCorr-IC: blue-□.

sidered to have converged if the final RMS point error was less than n_1 pixels after 30 iterations. We obtained these averages using, for each σ, n_2 randomly generated warps.

5.1. Experiments using the Takeo image

We started by reproducing to some extend the experimental setting of [5] using the Takeo image. We used $n_1 = 1$ pixel and, for each σ, $n_2 = 1000$ randomly generated warps. We assessed the performance of the forward additive and inverse compositional versions of our algorithm and the LK algorithm. We considered 3 cases. The first case was with no Gaussian smoothing prior to the calculation of image derivatives and no AWGN (Additive White Gaussian Noise). The second case was with smoothing but no AWGN. Finally, the third case was with both smoothing and AWGN of variance equal to 10 added to both the template and the target image. Fig. 3 shows the obtained average frequency of convergence.

As Fig. 3 (a) shows, for this experiment, the LK algorithms outperform the proposed methods. This is not un-



(a) (b) (c)

Figure 5. Average Frequency of Convergence vs Point Standard Deviation for Yale and AR databases. **Smoothing was used**. (a) Yale (b) AR-Occlusion (c) AR-Occlusion+illumination. LK-IC: black-◇. ECC-IC: yellow-△. IRLS-IC: cyan-x. GaborFourier-IC: red-*. GradientImages-IC: green-△. GradientCorr-IC: blue-□.

reasonable, as the affine distorted image was generated directly from the original image. In this case, there are no outliers, and as our algorithms remove some amount of information (most importantly the gradient magnitude), they inevitably perform worse. As Fig. 3 (b) illustrates, Gaussian smoothing improves the performance of all methods by providing a larger region of attraction. The performance gap between the LK and the proposed methods is now significantly smaller. Finally, as Fig. 3 (c) shows, if smoothing is used, none of the methods is affected too much by the AWGN even for a large noise variance (In fact, the performance of the LK methods is not affected at all). However, as next section shows, smoothing will not increase the robustness of methods which are not designed to be robust.

5.2. Experiments on the Yale and AR databases

In this section, we present our performance evaluation results obtained by using real image pairs (manually aligned), taken from the Yale B [20] and AR databases [19]. Our target was to assess performance in the presence of non-uniform illumination changes and occlusions. We used 100 different face pairs taken from the Yale database as follows. For each of the 10 subjects of the database we selected 1 template and 10 test images corrupted by extreme illumination changes. We also used 81 different face pairs taken from the AR database as follows. We selected 27 out of 31 subjects from the “dbf1” folder (4 subjects were discarded due to significant pose variation). For each subject, we selected 1 template image and 3 test images with sunglasses. Fig. 2 shows examples of images used in our experiments.

We used the average frequency of convergence for $\sigma = [1, 10]$ as the performance evaluation measure. We used $n_1 = 3$ pixels and, for each σ, $n_2 = 100$ randomly generated warps. Thus, for each σ, we used a total of 100×100 and 81×100 warps for Yale and AR respectively.

We assessed the performance of the inverse compositional versions of our algorithm (GradientCorr-IC), the LK algorithm (LK-IC) [5], the enhanced correlation (ECC-IC) algorithm [14], the iteratively re-weighted least squares algorithm (IRLS-IC) [11], and the Gabor-Fourier LK algorithm (GaborFourier-IC) recently proposed in [13]. The last two methods as well the mutual-information LK [12] (not considered here) are previously proposed robust methods. The implementations of the LK-IC and IRLS-IC algorithms are kindly provided by the authors. We implemented ECC-IC based on the forward additive implementation of ECC which is also kindly provided by the corresponding authors. Finally, we implemented GaborFourier-IC based on the implementation of LK-IC.

Additionally, based on the discussion in Section 3, we propose a new method: we used the orientation-based features of [24] and replaced regression with the inverse compositional algorithm. As gradients are treated exactly the same as intensities, we call this algorithm GradientImages-IC. We included this algorithm in our experiments to illustrate the performance improvement achieved by the proposed scheme which solves a continuous optimization problem based on the relation between gradients and intensities.

With the exception of GaborFourier-IC, for all methods, we considered two cases. The first case was with no Gaussian smoothing while the second one was with smoothing prior to the calculation of the image derivatives. We did not use smoothing for GaborFourier-IC as this is already incorporated in the method.

Figs. 4 and 5 show the average frequency of convergence for all face pairs and algorithms considered for the cases of “No Smoothing” and “Smoothing” respectively. Overall, the proposed GradientCorr-IC largely outperformed all other methods resulting in the most robust and stable performance. The performance improvement compared to GradientImages-IC is also more than evident. In particular, for large σ, GradientCorr-IC converged approximately

30-40$\%$ more frequently than GradientImages-IC. As Fig. 5 shows, Gaussian smoothing improved the performance of GradientCorr-IC and GradientImages-IC only. IRLS-IC seems to have worked well in the presence of occlusions but failed to converge when illumination changes were present. Surprisingly, Gaussian smoothing reduced the algorithm's performance. Although the results of [13] demonstrate that GaborFourier-IC is much more robust than the original LK-IC algorithm, our results show that this algorithm was also not able to cope with the extreme illumination conditions and occlusions considered in our experiments. Finally, the LK-IC and ECC-IC algorithms are not robust and, not too surprisingly, diverged for almost all face pairs considered.

6. Conclusions

We presented an efficient and robust approach to gradient ascent face alignment. Our method is based on the maximization of the gradient correlation coefficient and requires $O(nN)$ per iteration using the inverse compositional iterative procedure. Our experimental evaluation showed that, unlike state-of-the-art methods, our algorithm can cope with occlusions and severe non-uniform illumination changes. Thus, compared to state-of-the-art, the proposed scheme is equally fast, but significantly more robust.

References

[1] B.D. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," in *International joint conference on artificial intelligence*, 1981, pp. 674–679.

[2] G.D. Hager and P.N. Belhumeur, "Efficient region tracking with parametric models of geometry and illumination," *IEEE TPAMI*, vol. 20, no. 10, pp. 1025–1039, 1998.

[3] F. Dellaert and R. Collins, "Fast image-based tracking by selective pixel integration," in *Proceedings of the ICCV Workshop on Frame-Rate Vision*, 1999, pp. 1–22.

[4] S. Baker and I. Matthews, "Equivalence and efficiency of image alignment algorithms," in *CVPR*, 2001, pp. 1090–1097.

[5] S. Baker and I. Matthews, "Lucas-kanade 20 years on: A unifying framework," *IJCV*, vol. 56, no. 3, pp. 221–255, 2004.

[6] S. Baker, R. Gross, and I. Matthews, "Lucas-kanade 20 years on: Part 3," *Robotics Institute, Carnegie Mellon University, Tech. Rep. CMU-RI-TR-03-35*, pp. 1–51, 2003.

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

[8] B. Amberg, A. Blake, and T. Vetter, "On compositional Image Alignment, with an application to Active Appearance Models," in *CVPR*, 2009, pp. 1714–1721.

[9] R. Megret, J.B. Authesserre, and Y. Berthoumieu, "Bidirectional Composition on Lie Groups for Gradient-Based Image Alignment," *IEEE Transactions on Image Processing*, vol. 19, no. 9, pp. 2369–2381, 2010.

[10] M.J. Black and A.D. Jepson, "Eigentracking: Robust matching and tracking of articulated objects using a view-based representation," *IJCV*, vol. 26, no. 1, pp. 63–84, 1998.

[11] S. Baker, R. Gross, T. Ishikawa, and I. Matthews, "Lucas-kanade 20 years on: Part 2," *Robotics Institute, Carnegie Mellon University, Tech. Rep. CMU-RI-TR-03-01*, pp. 1–47, 2003.

[12] N. Dowson and R. Bowden, "Mutual information for lucas-kanade tracking (milk): An inverse compositional formulation," *IEEE TPAMI*, vol. 30, no. 1, pp. 180–185, 2007.

[13] A.B. Ashraf, S. Lucey, and T. Chen, "Fast Image Alignment in the Fourier Domain," in *CVPR*, 2010, pp. 2480–2487.

[14] G.D. Evangelidis and E.Z. Psarakis, "Parametric Image Alignment Using Enhanced Correlation Coefficient Maximization," *IEEE TPAMI*, pp. 1858–1865, 2008.

[15] G. Tzimiropoulos, V. Argyriou, S. Zafeiriou, and T. Stathaki, "Robust FFT-Based Scale-Invariant Image Registration with Image Gradients," *IEEE TPAMI*, vol. 39, pp. 1899–1906, 2010.

[16] V. Argyriou and T. Vlachos, "Estimation of sub-pixel motion using gradient cross-correlation," *Electronics Letters*, vol. 39, no. 13, pp. 980–982, 2003.

[17] AJ Fitch, A. Kadyrov, W.J. Christmas, and J. Kittler, "Orientation correlation," in *BMVC*, 2002, pp. 133–142.

[18] G. Tzimiropoulos and S. Zafeiriou, "On the Subspace of Image Gradient Orientations," *Arxiv preprint arXiv:1005.2715*, 2010.

[19] AM Martinez and R. Benavente, "The AR Face Database. CVC Technical Report# 24," 1998.

[20] A.S. Georghiades, P.N. Belhumeur, and D.J. Kriegman, "From few to many: Illumination cone models for face recognition under variable lighting and pose," *IEEE TPAMI*, vol. 23, no. 6, pp. 643–660, 2001.

[21] H.F. Chen, P.N. Belhumeur, and D.W. Jacobs, "In search of illumination invariants," in *CVPR*, 2002, pp. 254–261.

[22] A. Papoulis, S.U. Pillai, and S. Unnikrishna, *Probability, random variables, and stochastic processes*, McGraw-Hill New York, 2002.

[23] D. Hond and L. Spacek, "Distinctive descriptions for face processing," in *BMVC*, 1997, pp. 320–329.

[24] T.F. Cootes and C.J. Taylor, "On representing edge structure for model matching," in *CVPR*, 2001, pp. 1114–1119.

[25] Y. Ukrainitz and M. Irani, "Aligning sequences and actions by maximizing space-time correlations," *Computer Vision–ECCV 2006*, pp. 538–550, 2006.

[26] R. Gross, I. Matthews, J. Cohn, T. Kanade, and S. Baker, "Multi-pie," *Image and Vision Computing*, vol. 28, no. 5, pp. 807–813, 2010.