Toward an Efficient and Accurate AAM Fitting on Appearance Varying Faces

Hugo MERCIER[1], Julien PEYRAS[2], Patrice DALLE[1]

[1]IRIT - équipe TCI, UPS, 118 route de Narbonne
F-31062 Toulouse Cedex 9, FRANCE
{mercier,dalle}@irit.fr

[2]Dipartimento di Scienze dell'Informazione, via Comelico 39/41
I-20135 Milano, ITALIA
peyras@dsi.unimi.it

Motivations

- Use of *Active Appearance Models* within the *inverse compositional* framework [Baker & Matthews].
- Problem of appearance varying faces: fitting unknown faces or tracking appearance varying sequences.
- The best known solution (*simultaneous inverse compositional*) lacks efficiency.
- **Intention: Decrease the computational cost of the *simultaneous* algorithm.**
- The method test leads to a new definition of the ground truth shape.

AAM for facial modelling

- A facial AAM combines :
  1. a shape $\boldsymbol{s} = \boldsymbol{s}_0 + \sum_{i=1}^{n} v_i \boldsymbol{s}_i$,
  2. an appearance $A(x) = A_0(x) + \sum_{i=1}^{m} \lambda_i A_i(x)$.

  with the $\boldsymbol{s}_i$ and $A_i(x)$ variation modes obtained from a previously labelled image collection.
- Given initial parameters $[v_0, \lambda_0]$, the fitting goal is to find $[v, \lambda]$ that best models the face on an input image.

Original vs. proposed solution

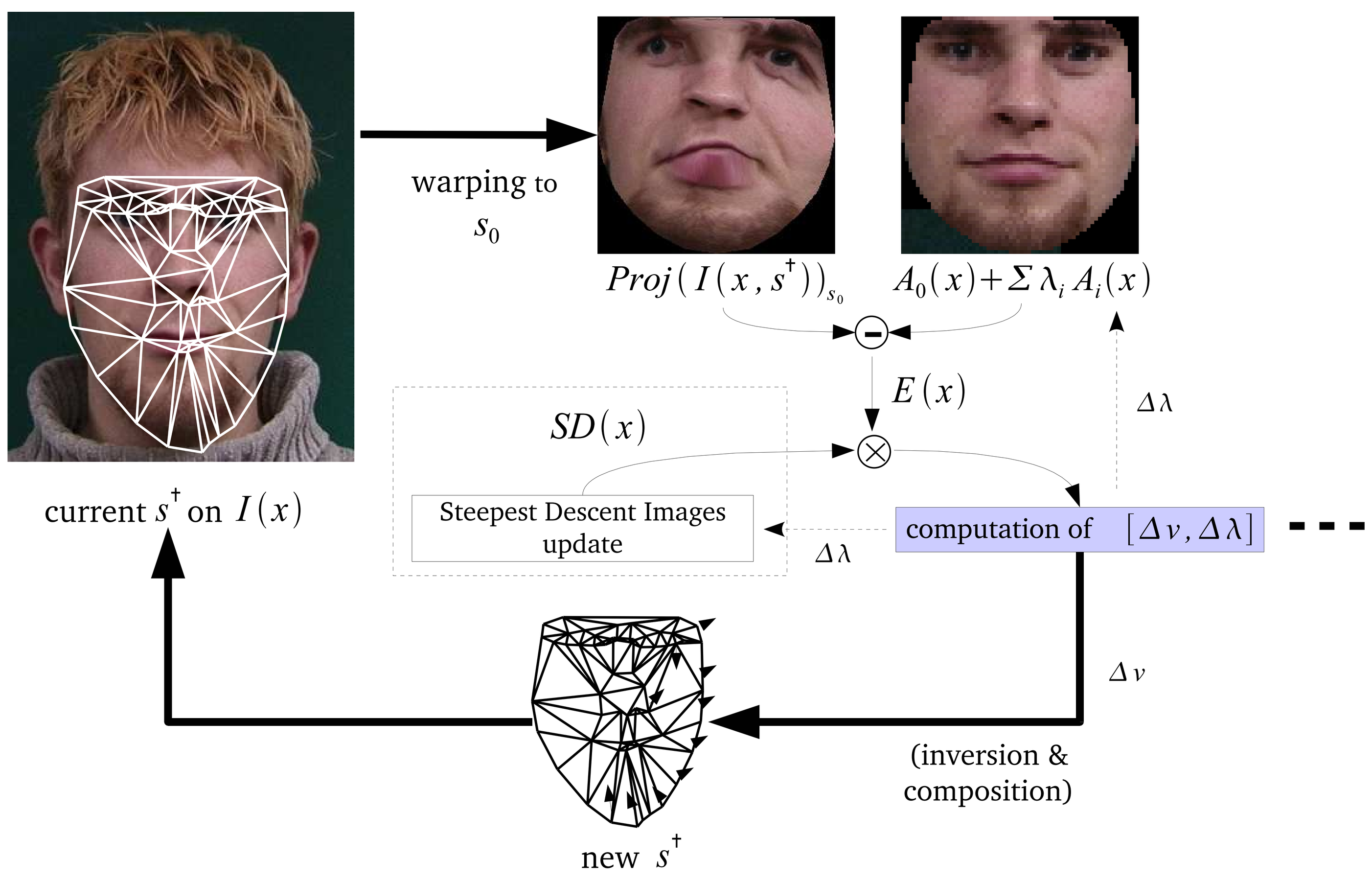


Illustration of the simultaneous inverse compositional algorithm

The original step, Hessian-based [Baker & Matthews]

$$[\Delta \boldsymbol{v}, \Delta \boldsymbol{\lambda}]^T = -\boldsymbol{H}^{-1} \sum_x SD^T(x) E(x)$$

where

$$\boldsymbol{H} = \sum_x SD(x)^T SD(x)$$

and is computed in $O((n+m)^2 N)$
for n shape vectors, m appearance vectors and a s_0 image resolution of N pixels.

The proposed computation, regulation based

$$[\Delta \boldsymbol{v}(t), \Delta \boldsymbol{\lambda}(\boldsymbol{t})]^T = -\boldsymbol{C}(t-1) \odot \sum_x SD^T(x) E(x)$$

The c_i coefficients are computed in the following manner:

for $i = 1$ to $n+m$ **do**
  if $\Delta\omega_i(t-1)\Delta\omega_i(t) > 0$ **then**
    $c_i(t) \leftarrow c_i(t-1)\eta_{inc}$
  else
    $c_i(t) \leftarrow c_i(t-1)/\eta_{dec}$
  end if
end for

where the computation is negligible compared to $O((n+m)^2 N)$.
$\Delta\omega_i$ stands for either Δv_i or $\Delta\lambda_i$. The parameters η_{inc} and η_{dec} are empirically fixed.

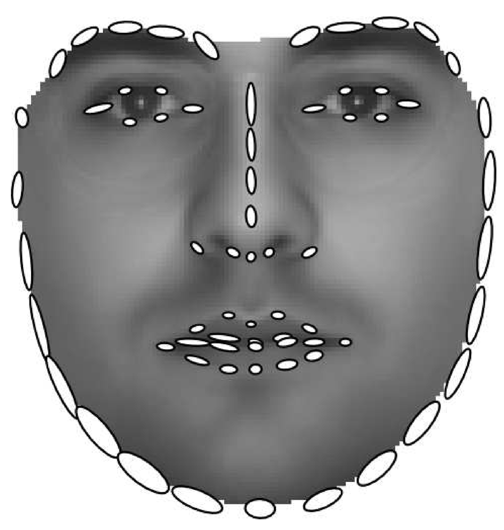
Evaluation protocol

- Performance comparison between the Hessian-based algorithm and our version.
- Test of two fitting features on both known and unknown frontal neutral faces: **accuracy** and **efficiency**.

- Introduction of a statistical-based method to build the ground truth data. Each face has been manually labelled 11 times.
- Score a labelling with respect to the variance of each vertex coordinates.

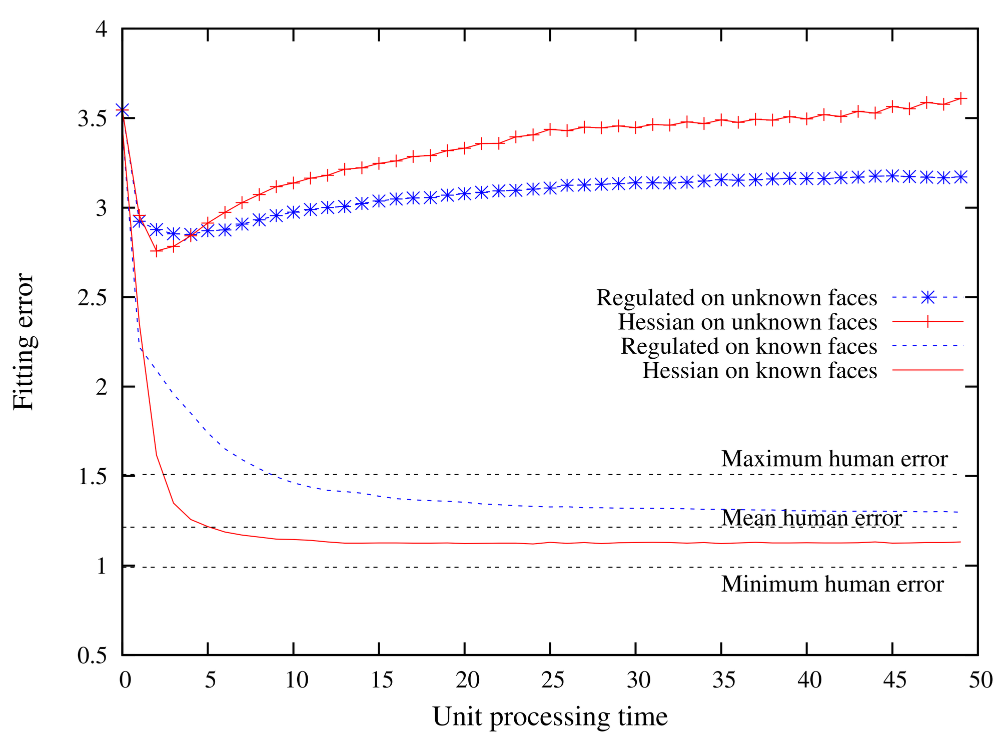
The fitting error $e_i(\boldsymbol{s})$ of a shape $\boldsymbol{s}$ on an image i, is defined by the average of the Mahalanobis distances between the obtained vertex location s_v and its ground truth definition $\mu_{i,v}$, for all n_V vertices:

$$e_i(\boldsymbol{s}) = \frac{1}{n_V} \sum_{v=1}^{n_V} \sqrt{(\boldsymbol{s}_v - \boldsymbol{\mu}_{i,v})^T \boldsymbol{\Sigma}_v^{-1} (\boldsymbol{s}_v - \boldsymbol{\mu}_{i,v})}$$



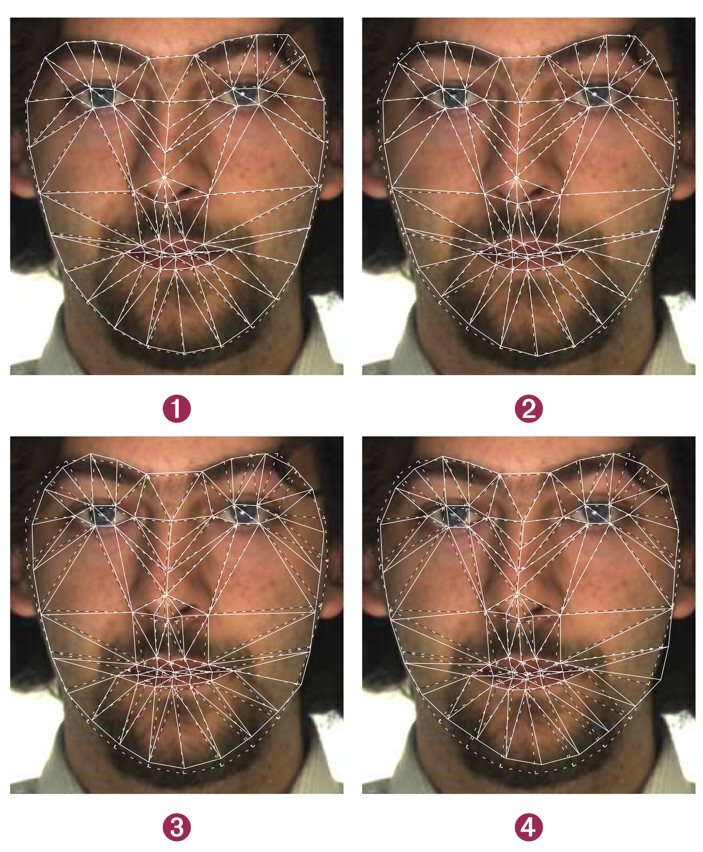
Representation of the covariance Σ_v by an ellipse, for each vertex, here displayed on the mean face

Results



Fitting error evolution accross time.

- Iteration time is different for the regulated (faster) and the Hessian-based. Algorithm performances are thus compared at same units of processing time.
- In the known faces test, the Hessian-based algorithm performs better than the regulated, as it reaches faster a lower minimum.
- In the unknown faces test, minima are reached after an equivalent processing time for the two algorithms. The fitting quality is almost equivalent.



❶ and ❷ are typical fittings obtained on known faces by the Hessian-based and the regulated algorithms.

❸ and ❹ are the best fittings obtained on unknown faces for both the Hessian-based and the regulated.

Future works

- In the unknown faces test, the rise of fitting error is due to the inability of algorithms to deal with non-Gaussian noise. We will investigate on the use of a robust error function.
- The processing time to reach a minimum has to be compared for different values of n, m and N.
- It has to be compared to other variants of the inverse compositional algorithm, particularly the steepest descent minimization and the diagonal Hessian approximation.