Universidad de Alcalá
BIBLIOTECA

Document downloaded from the institutional repository of the University of Alcala: http://ebuah.uah.es/dspace/

This is a preprint version of the following published document:

Nuevo, J., Bergasa, L.M. & Jiménez, P. 2010, "RSMAT: Robust Simultaneous Modeling and Tracking", Pattern Recognition Letters, vol. 31, no. 16, pp. 2455-2463

Available at http://dx.doi.org/10.1016/j.patrec.2010.07.016

© 2010 Elsevier

(Article begins on next page)

This work is licensed under a

Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License.

RSMAT: Robust Simultaneous Modeling and Tracking

Jesus Nuevo, Luis M. Bergasa*, Pedro Jiménez

Department of Electronics, University of Alcala. Esc. Politecnica, Crta Madrid-Barcelona, Km 33,600. 28871 Alcala de Henares, Madrid

Abstract

This paper describes a robust on-line appearance modeling and tracking method, based on Simultaneous Modeling and Tracking (SMAT). The appearance model is defined by a series of clusters, built in a video sequence using previously encountered samples. This model is used to search for the corresponding element in the following frames. Three alternative incremental clustering methods are proposed to increase the robustness and description capabilities of the model. The proposal is evaluated on an application of face tracking for driver monitoring. The test set comprises sequences of drivers recorded outdoors and in a truck simulator, which contain examples of occlusions and self-occlusions, as well as illumination changes. The performance is evaluated and compared with that of the original SMAT proposal and the recently presented Stacked Trimmed Active Shape Model (STASM). Our proposal shows better results than the original SMAT and similar fitting error levels to STASM, with much faster execution times and better robustness to self-occlusions.

Keywords: Incremental clustering, appearance modeling, face tracking, robust fitting, real-time, driver monitoring

1. Introduction

Morphable object tracking is a very active research field in computer vision. This paper presents a method to model and track deformable objects simultaneously, using incremental clustering to model the appearance of the object. Our proposal is a modification of the Simultaneous Modeling and Tracking (SMAT) method proposed in (Dowson and Bowden, 2005). As the texture model is built as the video sequence progresses, *a priori* learning of the model is avoided. An active model characterizes the shape that relates the position of different features.

Human face is one of the most commonly used examples of deformable objects, and a comprehensive number of methods and applications have been developed (Yang et al., 2002). Face location and tracking are the first processing stages of most computer vision systems for driver monitoring. The method presented in this paper will serve as the base for an inattention detection system such as the one presented by the authors in (Bergasa et al., 2006). Because this system has to work with any driver under many different outdoor scenarios, robustness is of critical importance. The tracker must continue working when head turns, partial occlusions and illumination changes take place, in both day and night scenarios. The incremental nature of the proposed algorithm allows for the creation of a particular model for each driver in an online process.

Several systems to monitor drivers using computer vision have already been introduced. A few commercial systems are available as accessories for their instalation in vehicles (Seeing Machines, 2004, 2007; SmartEyeAG, 2009). There are very few details available in the literature regarding the methods and parameters of these systems. Cost and reliability are still obstacles for automakers to offer this kind of system as an integrated option. Most use one or two cameras to track the head and eyes of the subject (Matsumoto and Zelinsky, 2000; Victor et al., 2001; Kutila, 2006).

The rest of the paper is structured as follows. Section 2 presents a few remarkable works in face tracking in the literature that are related to our proposal. Section 3 describes the original SMAT approach, which is discussed in section 4. Then, improvements over the original method are described. Section 6 presents the experimental results. This paper closes with conclusions and future work.

2. Background

Human face tracking is of the greatest interest for many applications, as vast amounts of information are contained in face movements, features and gestures. This information is constantly used for human non-verbal communication. Facial expression systems (Belhumeur et al., 1997; Buenaposada et al., 2008) often use face tracking as a first processing step. General purpose template-based trackers have been used to track non-rigid objects with success. Several efficient approaches have been presented (Hager

*Corresponding author: Tel. (+34) 91885 6569, Fax: (+34)91885 6591

Email addresses: jnuevo@depeca.uah.es (Jesus Nuevo), bergasa@depeca.uah.es (Luis M. Bergasa), pjimenez@depeca.uah.es (Pedro Jiménez)

and Belhumeur, 1998; Buenaposada et al., 2006; Jurie and Dhome, 2002; Baker and Matthews, 2001).

Statistical models have been widely used for face modeling and tracking. Active Shape Models (Cootes et al., 1995) (ASM) are similar to the active contours (*snakes*), but include constraints from a Point Distribution Model (PDM) (Dryden and Mardia, 1998) computed in advance from a training set. Advances in late years have increased their robustness and precision to remarkable levels (STASM (Milborrow and Nicolls, 2008)). Extensions of ASM that include modeling of texture have been presented, of which Active Appearance Models (AAMs) (Cootes et al., 2001) are arguably the best known. Constrained Local Models (Cristinacce and Cootes, 2006; Wang et al., 2008) improve the results of holistic approaches such as AAMs by using localized models and searching. All these models have an offline training phase, which require comprehensive training sets so they can generalize properly to unseen instances of the object. This is a time consuming and error prone process. Several proposals have been presented recently that aim at addressing this problem (Cootes, 2005; Tong et al., 2009).

Methods that skip the *a priori* training step have been presented in the literature. Most of them focus on patch tracking on a video sequence. The classic approach is to use the image patch extracted on the first frame of the sequence as a template to search for similar patches on the following frames. Lukas-Kanade method (Lucas and Kanade, 1981) was one of the first proposed solutions and it is still widely used. Jepson *et al.* (Jepson et al., 2003) presented a system with appearance model based on three components: a stable component that is learned over a long period based on wavelets, a 2-frame tracker and an outlier rejection process. Matthews *et al.* (Matthews et al., 2004) proposed an *strategic update* of the template, which keeps the template from the first frame to correct errors that appear in the localization. When the error is too high, the update is blocked. Segvic (Segvic et al., 2006) proposed a solution with fixed template and an adaptative window around the features. Yin and Collins (Yin and Collins, 2007) build an adaptive view-dependent appearance model on-line. The model is made of patches selected around Harris corners, and model and target patches are matched using correlation. Changes in position, rotation and scale are obtained with the Procrustes algorithm (Gower, 1975).

Another successful line of work in object tracking without *a priori* training is based on classification instead of modeling. Collins and Liu (Collins et al., 2005) presented a system based on background/foreground discrimination. Avidan (Avidan, 2007) presents one of the many systems that use machine learning to classify patches (Grabner and Bischof, 2006; Pham and Cham, 2007). Avidan uses weak classifiers trained every frame and AdaBoost to combine them. Pilet *et al.* (Pilet et al., 2005) train keypoint classifiers using Random Trees that are able to recognize hundreds of keypoints in real-time.

Simultaneous Modeling and Tracking (SMAT) is in line with methods like Lucas-Kanade and the *strategic update*, relaying on matching to track patches. SMAT however builds a more complex model based on incremental clustering. A similar clustering technique has been used in (Cristinacce and Cootes, 2008) for groupwise registration. SMAT has been extended recently to use linear displacement predictors (Ellis et al., 2007) for faster tracking. (Jimenez et al., 2009) used SMAT to model texture in a 3D face tracking system. We have chosen SMAT for our face tracking application because the model is adaptative and it is likely to comply with the requirements listed above. However, SMAT has a few limitations, and that is why we propose some modifications over the original proposal to obtain a more robust model and better tracking. We name the new approach *Robust SMAT*.

3. Simultaneous Modeling and Tracking

Simultaneous Modeling and Tracking (SMAT) tries to model both the appearance of a series of features and how their positions are related (the *shape*), and doing so without a pre-trained model. The appearance and shape models are independent, and fitted independently on each frame: features are tracked with their respective appearance models. Their final positions are then processed using the shape model. If the positions are considered correct and not due to tracking errors, the models are updated. If not, the data corresponding to that frame are discarded. Figure 1 shows a flow chart of the algorithm.

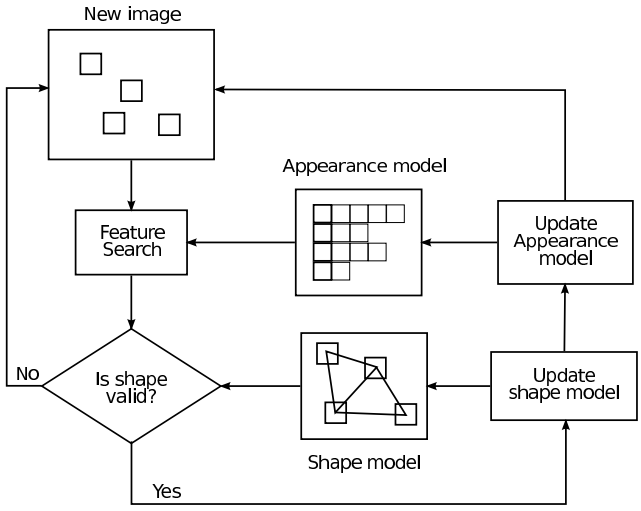


Figure 1: SMAT block diagram

In this section, we briefly describe the appearance and shape model of SMAT.

3.1. Appearance Modeling

SMAT builds a library of exemplars obtained from previous frames, image patches in our case. Similar patches will be close together in the feature space, away from other

points representing dissimilar appearances. These groups, or clusters, can be used to represent the appearance of the object. Dowson and Bowden assumed that the points in each cluster in feature space follow a Gaussian distribution, and clusters can be defined by their median and variance. The median is more robust than the mean, and the later may result in a smoothed patch that may not correspond to any possible appearance.

The clusters are updated incrementally as new samples become available. The membership $m_k(x)$ of a new patch to a cluster k will depend on the relative distance $d(x, \mu_k)$ to the median (or *representative*) μ_k, and a threshold relative to the variance of the cluster, $\tau(\sigma_k)$. In later works (Dowson and Bowden, 2006), the condition for membership depends on the ratio of probabilities of the exemplar x belonging to the foreground (a cluster with representative μ_k) and to the background

$$\frac{p(fg\,|d(x,\mu_k),\sigma_{fg_k})}{p(bg\,|d(x,\mu_k),\sigma_{bg_k})} \tag{1}$$

where σ_{fg_k} is obtained from the distances between the representative and the other exemplars in the cluster, and σ_{bg_k} is obtained from the distances between the representative and the exemplars in the cluster offset by 1 pixel.

Clusters are assigned a weight that increases when they are frequently updated. For each new frame, the weight is calculated as

$$w_k^{(t+1)} = \begin{cases} (w_k^{(t)} + \alpha)\frac{1}{1+\alpha} & if \quad k = k_u \\ w_k^{(t)}\frac{1}{1+\alpha} & \text{otherwise} \end{cases} \tag{2}$$

where $\alpha \in [0, 1)$ is the learning rate, and k_u is the index of the updated cluster. If the new patch is does not belong to any cluster, a new cluster is created, with the new patch as its only member, with weight $w_k^{(t+1)} = 0$. Up to K clusters are kept, and the cluster with lowest weight is removed if K is reached. To limit memory requirements, the number of samples kept in a cluster is also limited to M.

3.2. Point Distribution Model

SMAT point distribution model is built in a similar fashion to that of the appearance. Clusters are formed with shapes that are close in the point distribution space. The mean of the shapes in each cluster is used as representative. Mahalanobis distance is used to compare the shape resulting from the independent feature tracking with each representative:

$$d_k(\mathbf{s}) = [(\mathbf{s}' - \bar{\mathbf{s}}_k)^T \mathbf{\Sigma}_k^{-1} (\mathbf{s}' - \bar{\mathbf{s}}_k)]^{\frac{1}{2}} \tag{3}$$

where $\mathbf{s}'$ is the shape $\mathbf{s}$ translated to the origin, $\bar{\mathbf{s}}_k$ is the representative of cluster k also translated to the origin, and $\mathbf{\Sigma}_k$ is the covariance matrix of the data in the cluster. As $\mathbf{\Sigma}_k$ will often be non-invertible, singular value decomposition (SVD) is used to obtain its eigenvalues $\mathbf{D}$ and eigenvectors $\mathbf{A}$, and the distance is computed as

$$d_k(\mathbf{s}) = [\mathbf{A}(\mathbf{s}' - \bar{\mathbf{s}}_k)]\mathbf{D'}^{-1} \tag{4}$$

where $\mathbf{D'}^{-1}$ is the pseudo-inverse of $\mathbf{D}$.

3.3. Enforcing point distribution model constraints

Once a shape $\mathbf{s}$ has been found to belong to a cluster k, the positions are constrained by projecting the shape over the eigenvectors of the cluster and limiting the coefficient values to the eigenvalues in $\mathbf{D}$. The constrained shape $\mathbf{s}'_c$ is computed as

$$\mathbf{s}'_c = \mathbf{A}min(\mathbf{A}^T(\mathbf{s}' - \bar{\mathbf{s}}_k), \mathbf{D}^{-\frac{1}{2}}) + \bar{\mathbf{s}}_k \tag{5}$$

If the distance between the position of a feature in $\mathbf{s}'$ and $\mathbf{s}'_c$ is greater than δ, the appearance model of this feature is not updated to prevent introducing outliers. Also, for the next frame the feature position is reset to its value in $\mathbf{s}'_c$.

In the case that the shape $\mathbf{s}$ does not belong to any existing cluster, a new one is created in the point distribution model only if less than 25% of the feature trackers have created new clusters in their models. Introduction of novel data in both types of models at the same time indicates a possible tracking loss, and it is blocked for increased robustness.

4. Discussion

The defining characteristic of SMAT is its incremental model. Allowing for the model to evolve as the object changes helps create a robust and highly specific model. On the downside, the absence of prior information makes the algorithm specially sensible to the first frames of the sequence. Still, this is an improvement over previous works (Matthews et al., 2004)(Kaneko and Hori, 2002) that relayed on the first frame alone.

Incorrect initialization in the first frames introduces undesirable exemplars in the model, creating clusters that would lead to further tracking of erroneous characteristics. This *model tainting* also happens to a lesser degree when tracking is lost: new clusters with exemplars from the background may be created. See for example figure 2. Tracking is lost due to total occlusion in figure 2(b), and then correctly repositioned. However, some clusters in 2(c) now have a representative that is a bright white exemplar, which is not a possible texture of the corresponding patch.

The representative in a cluster can change as the model evolves. This feature could, however, result in overlapping of the clusters as they move in the feature space, wasting memory space and reducing the quantity of the space that can be modeled by the K clusters. In our experiments, we have observed that the opposite situation occurs much more frequently: the representative of the cluster rarely changes after the cluster has reached a certain number of elements.

(a) Before tracking loss (b) Tracking loss (c) Model fitted after tracking loss

Figure 2: An example of model tainting. Red crosses indicate points for which the search diverged. Each row of squares on the right hand side is the group of clusters that represents the texture of the patch around a landmark. For example, the first two rows model the appearance of the eye's pupil, and the third and fourth the corners of the mouth. Each small square shows the representative of that particular cluster. The most recently updated cluster is marked with a red dot.

Other problem of the clustering used by SMAT is overfitting. If very similar patches are constantly introduced, one of them will be chosen as representative of one of the clusters, and as the size limit M is reached, other distinct exemplars will be discarded, reducing the variance of the cluster. At a frame rate of 25 fps, for $M = 50$, the cluster could overfit in less than 2 seconds. This procedure would discard valuable information and future, subtle changes to the feature will lead to the creation of another cluster.

The shape modeling of SMAT also suffers from these problems. Additionally, the process of comparing the shapes as described above does not remove the rotation and scale from the shape. This procedure is usually carried out with Procrustes method (Gower, 1975). A slightly rotated shape can have a significantly different Mahalanobis distance, leading to the generation of new clusters for shapes that are actually similar or blocking updates.

To overcome some of these problems, we propose a variation of the original SMAT, described in the next chapter.

5. Robust SMAT

Incremental clustering deals with the problem of clustering a set of points in space that are presented sequentially, maintaining a set of clusters as optimal as possible in some sense. Most research in this field is related to document retrieval, database processing (Can, 1993) and data mining in general. A few approaches are mentioned here, we refer the reader to literature reviews on clustering for more details (Jain et al., 1999; Rasmussen, 1992). Chapters 16 and 17 in (Manning et al., 2008) present clustering in the field of information retrieval.

In a previous work (Jimenez et al., 2009), the shape model of SMAT was replaced by a rigid 3D shape model, created on-line with a stereo camera pair. We have used in this paper an alternative method for modeling the shape, using a pre-learned shape model similar to that of Active Shape Models, fitted with Huber M-estimator (Huber, 1981). Using a robust fitting function has showed improved performance in several published works (Rogers and Graham, 2002). This chapter presents three alternative clustering methods for modeling the appearance.

Arguably the simplest and most frequently used incremental clustering method is the leader algorithm (Hartigan, 1975; Spath and Bull, 1980). Each cluster $\mathcal{C}_i$ is defined by only one exemplar, and a fixed membership threshold T. If an incoming exemplar fulfills a condition (usually being within a distance of the representative), it is marked as member of that cluster, otherwise it becomes a cluster on its own. This algorithm has been extended to work with fuzzy clusters (Asharaf and Murty, 2003), and interval data (Asharaf et al., 2006).

The *leader* method is the first clustering method tested in this work. Its main benefits are its low complexity ($O(nk)$) and the low memory requirements ($O(k)$, where n is the dimension of the exemplars and k is the number of clusters) (Jain et al., 1999). Because only the cluster representatives are stored, many more clusters can be kept in memory. In order to remove clusters generated by outliers, the maximum number of clusters has been limited.

It was referred above that the representative of the clusters in SMAT rarely changes after a few exemplars have been added. A modification of the leader algorithm has also been tested, where instead of making the first exemplar the representative and only member of a newly created cluster, the first few exemplars added to the cluster are kept, up to P. We will refer to this method as *leader*P, described in algorithm 1.

The median of the cluster is chosen as the representative, as in section 3.1. When the number of exemplars in the cluster reaches P, all exemplars but the representative are discarded, and the *leader* algorithm is used from then on. P is chosen as a small number (we use 10). The membership threshold is however flexible, and its value is computed from the distances between the representative and each of the exemplars that are found to be members of the cluster. Because the representative is fixed, the exemplars can be discarded while keeping many more values of the distances in memory. Note that this is not possible in the original SMAT method, as the values become invalid when the representative changes.

Finally, a hierarchical method is proposed. Exemplars are added to the clusters as in the original clustering in SMAT, and are sub-clustered again using the *leader*P al-

Algorithm 1 LeaderP clustering

Let $C = \{\mathcal{C}_1, \ldots, \mathcal{C}_n\}$ be a set of n clusters, with weights $\{w_1^t, \ldots, w_n^t\}$ and membership thresholds $\{T_1, \ldots, T_n\}$

procedure LEADERP(E, C) ▷ cluster patch E
  for all $\mathcal{C}_i$ **do** ▷ $R_{\mathcal{C}_i}$ is the representative of $\mathcal{C}_i$
    if $d(R_{\mathcal{C}_k}, E) < T_i$ **then** ▷ $E \in \mathcal{C}_k$
      UPDATEWEIGHTS($w_1^t, \ldots, w_n^t$) as in eq. 2
      if C_i fixed **then**
        $d_j \leftarrow d(R_{\mathcal{C}_k}, E)$ ▷ $d_r,\ r = 0, \ldots, r-1$ contain past distances
        $T_k \leftarrow \tau(Sdv(d_1, \ldots, d_k))$
      else
        $R_{\mathcal{C}_k} \leftarrow median(\mathcal{C}_k)$
        if $|\mathcal{C}_k| = P$ **then**
          FIXREPRESENTATIVE($\mathcal{C}_k$) ▷ Discard all patches but the median
        end if
      end if
      return
    end if
  end for
  Create new cluster $\mathcal{C}_{n+1}$, with E as representative.
  Set $w_{n+1}^{t+1} \leftarrow 0$
  $C \leftarrow C \cup \mathcal{C}_{n+1}$
  if $n + 1 > K$ **then** ▷ Remove the cluster with lowest weight
    Find $\mathcal{C}_k \mid w_k \leq w_i \quad i = 1, \ldots, n$
    $C \leftarrow C \setminus \mathcal{C}_k$
  end if
end procedure

Algorithm 2 Hierarchical clustering

Let $C = \{\mathcal{C}_1, \ldots, \mathcal{C}_n\}$ be a set of n clusters, with weights $\{w_1^t, \ldots, w_n^t\}$ and membership thresholds $\{T_1, \ldots, T_n\}$
Let $S^j = \{\mathcal{S}_1^j, \ldots, \mathcal{S}_m^j\}$ be the set of subclusters of a cluster $\mathcal{C}_j$

procedure HIERARCHICAL(E, C) ▷ cluster patch E
  for all $\mathcal{C}_i$ **do** ▷ $R_{\mathcal{C}_i}$ is the representative of $\mathcal{C}_i$
    if $d(R_{\mathcal{C}_k}, E) < T_i$ **then** ▷ $E \in \mathcal{C}_k$
      UPDATEWEIGHTS($w_1^t, \ldots, w_n^t$) as in eq. 2
      LEADERP(E, S^k)
      $R_{\mathcal{C}_k} \leftarrow median(\mathcal{C}_k)$
      T_k
    end if
    return
  end for
  Create new cluster $\mathcal{C}_{n+1}$, with E as representative.
  Set $w_{n+1}^{t+1} \leftarrow 0$
  if $n + 1 > K$ **then** ▷ Remove the cluster with lowest weight
    Find $\mathcal{C}_l \mid w_l \leq w_i \quad i = 1, \ldots, n$
    $C \leftarrow C \setminus \mathcal{C}_l$
  end if
end procedure

gorithm, with a smaller value of P. This hierarchical structure is similar to the Leader-Subleader algorithm in (Vijaya et al., 2004). Our method, however, builds the clusters and sub-clusters simultaneously, and does not require several passes over the exemplar set. This method is summarized in algorithm 2.

Using a hierarchy helps reduce the memory requirements and the number of distances to be computed, while the risk of overfitting is diminished. If many, very similar patches were included, they would be grouped in one of the sub-clusters, and the other sub-clusters would remain unmodified. The representative of the cluster can shift from sub-cluster to sub-cluster.

6. Tests and results

This section presents the video sequences used to test different clustering methods. A comparison between our approach and the original SMAT is presented then.

6.1. Test set

We have used the RobeSafe Driver Monitoring Video (RS-DMV) dataset to perform the tests.

This test set features subjects driving in a vehicle moving outdoors, and in a truck simulator. Such scenarios present some challenges for a face tracking algorithm: they involve constant movements, head rotations (self-occluding part of the face) or partial occlusions by hands, scarfs or other elements such as glasses. Drivers also talk and gesture frequently. The environment changes as the vehicles move, and with it change the background and the number and position of the light sources and shadows produced by trees or buildings.

The RS-DMV dataset contains 10 sequences, 7 recorded outdoors (*Type A*) and 3 in a simulator (*Type B*). Outdoor sequences were recorded on RobeSafe's vehicle moving at the campus of the University of Alcala. Weather conditions during the recordings were mostly sunny, which made noticeable shadows appear on the face.

Type B sequences were recorded in a realistic truck simulator. The recordings took place in a low-light scenario that approached nighttime conditions. Motion blur appears during movements, due to the long exposure time of the cameras. A few images from the sequences can be seen in figure 3.

The outdoor sequences are around 2 minutes long, and sequences in the simulator are close to 10 minutes in length. Images were captured at high resolution in grayscale, at 30 frames per second. The algorithms in this paper, however, were tested on images of approximately 320×240 pixels.

The RS-DMV is publicly available, free of charge, for

Figure 3: Samples of test sequences

research purposes. More information on this dataset and how to obtain it can be found at the author's website[1].

6.2. Performance evaluation

Videos in the RS-DMV dataset were marked by a human operator, at a rate of approximately 1 frame per second. The error values are computed only for these *keyframes*. Twenty points are marked over the face.

The metric m_e, introduced by Cristinacce and Cootes (Cristinacce and Cootes, 2006), is invariant to the size of the face, and used to evaluate the distance between ground-truth $\mathbf{x}_i$ and the estimated position of the points $\hat{\mathbf{x}}_i$.

$$m_e = \frac{1}{ns}\sum_{i=1}^{n} d_i, \qquad d_i = \sqrt{(\mathbf{x}_i - \hat{\mathbf{x}}_i)^T(\mathbf{x}_i - \hat{\mathbf{x}}_i)} \qquad (6)$$

where n is the number of points and s is the inter-ocular distance. The points on the chin and the exterior of the eyes are discarded, because their location changes much from person to person. Because only 17 points are used, we note the metric as m_e17. The value of m_e17 for a keyframe is set to ∞ if tracking is lost, or if the face cannot be found.

6.3. Results

We tested the performance of our R-SMAT approach on the RS-DMV dataset, as well as the original SMAT proposal.

(R-)SMAT needs to be properly initialized due to the absence of an a priori model, because small errors in the first frame would affect the rest of the test. To minimize this problem, the shape is initialized by hand. At the end of this section, the performance of R-SMAT automaticaly initialized with STASM (Milborrow and Nicolls, 2008) is presented.

First, we study the influence of the size of the features used. Because each type of sequences has its own characteristics, results are presented for outdoor and simulator sequences separatedly.

Three patch sizes were tested: 11×11, 15×15 and 20×20. Figure 4 shows the results of the three different sizes, in all cases using *leaderP* clustering. The graphs correspond to all keyframes. All three patch sizes show similar results, which are nearly identical for *type B* sequences. However, sizes of 15×15 and 20×20 perform better than 11×11 for *type A* sequences. The difference comes from the amount of lost frames, which is clear in table 1. One of the main characteristics of *type A* sequences is frequent head movements, and when those happen motion blur appears. Small features are more easily lost, and thus working with a larger patch improves the chances of success.

Table 1: Track losses for different patch sizes in pixels

Size	Sequences	Mean	Max	Min
11×11	*Type A*	5.87%	12.5%	3.47%
	Type B	0.76%	1.14%	0%
15×15	*Type A*	0.99%	2.08%	0%
	Type B	1.13%	1.96%	0%
20×20	*Type A*	0.33%	1.43%	0%
	Type B	1.52%	4.58%	0%

Figure 5 shows the performance of the original SMAT clustering compared with the proposed clustering algorithms. All models use patches of 15×15 pixels.

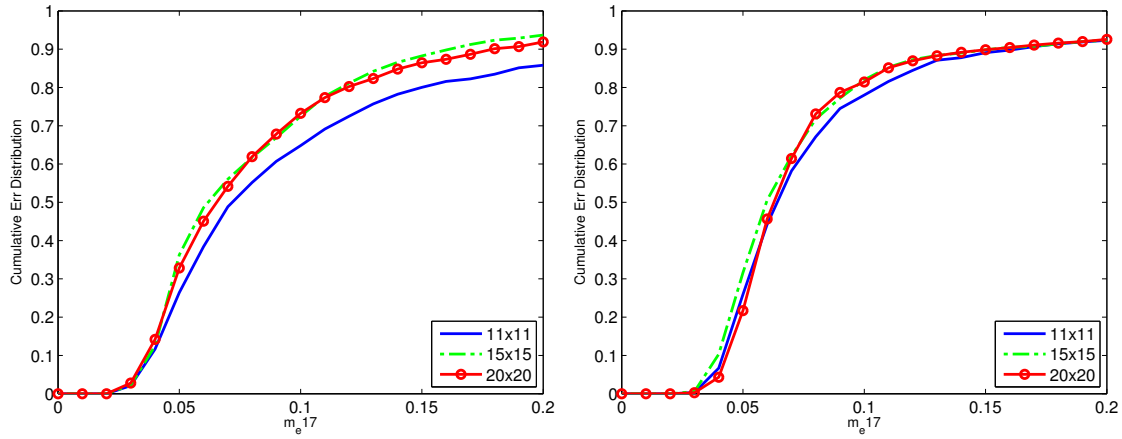
All three alternative methods present much better performance than the original SMAT clustering. This is specially clear in figure 5(b). Table 2 shows the tracking losses for the four clustering methods, as a percentage of the *keyframes* in the sequences. For comparison, results for STASM have also been included. The high number of tracking losses for this method is due to it not being able to fit to the face when it is turned.

That such a simple clustering as *leader* can perform as good as more complicated methods indicates that the texture of the patches can be modeled with a small number of exemplars. Other scenarios, with stronger deformations or changes in appearance may require more complicated clustering, and methods like *leaderP* and *hierarchical* would then outperform *leader*. Results for R-SMAT presented hereinafter were obtained using *leaderP*.

Results above have been obtained initializing SMAT and R-SMAT with landmarks from the handmarked ground-truth data. In a real scenario, an automatic algorithm would be used to initialize SMAT and R-SMAT.

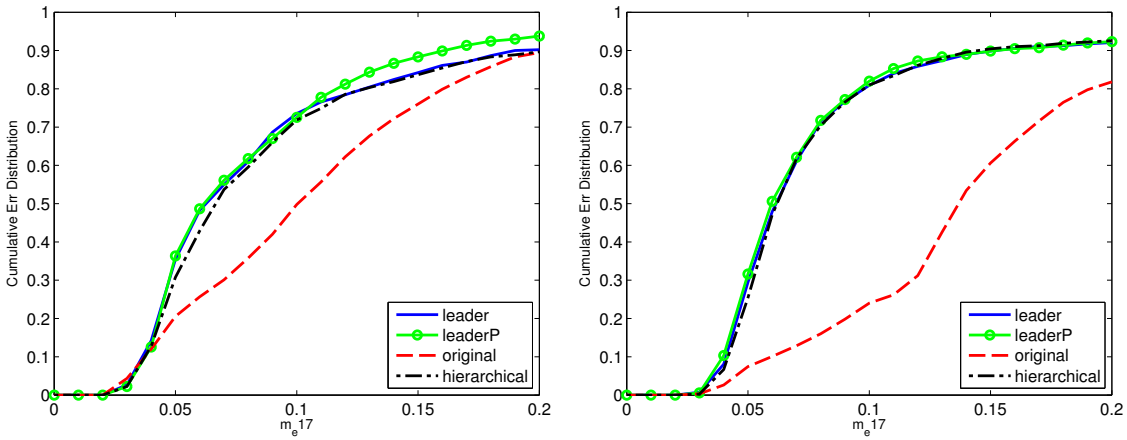
We have used STASM for this task. STASM is very accurate when the face is frontal to the camera. It does not work in real-time, but a one-time delay is acceptable.

[1] www.robesafe.com/personal/jnuevo



(a) *Type A* (b) *Type B*

Figure 4: Comparison of the performance of different patch sizes, with *leaderP* clustering



(a) *Type A* (b) *Type B*

Figure 5: Comparison of the performance of clustering algorithms

Table 2: Track losses for different clustering methods and types of sequences

Method	Sequences	Mean	Max	Min
leader	*Type A*	5.21%	18.05%	0%
	Type B	0.81%	2.12%	0%
leaderP	*Type A*	0.99%	2.08%	0%
	Type B	0.71%	1.96%	0%
hierarchical	*Type A*	3.65%	10.6%	0%
	Type B	0.32%	0.82%	0%
SMAT	*Type A*	1.77%	5.03%	0%
	Type B	2.01%	3.60%	0%
STASM	*Type A*	5.43%	15.0%	0%
	Type B	1.92%	3.27%	0%

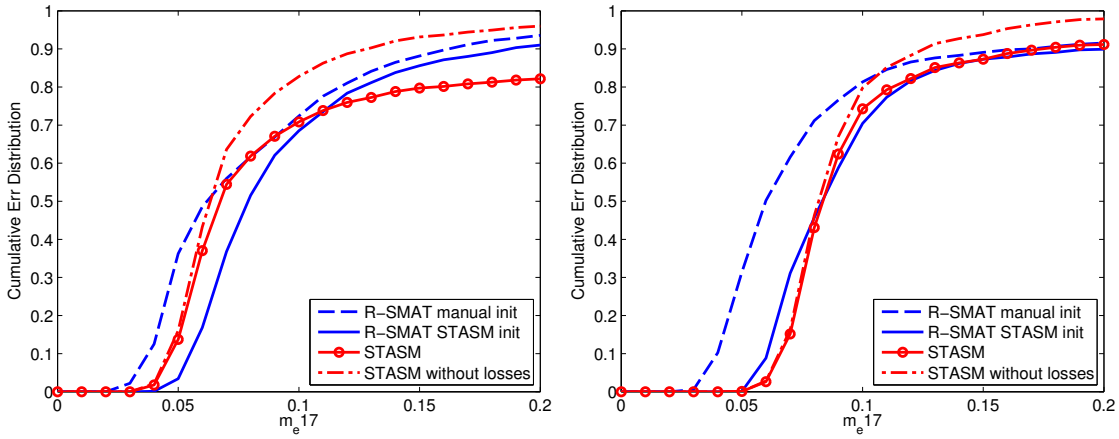
STASM was run on the first frame of each sequence, and its estimation used to initialize the position of R-SMAT in that video. Figure 6 presents the error distributions of R-SMAT with manual and automatic initalization. The error of STASM is also shown.

The results obtained with automatic initialization are worse than for manual initialization. The loss of accuracy is acceptable, with a 5% loss at $m_e17 = 0.1$ for *type A* sequences, and 10% loss for *type B*. The mean of the error of STASM in the first frame is 0.0571 for *type A* sequences and 0.0805 for *type B*, which are close to the increases in error.

A few samples of R-SMAT fitted to the face of drivers can be seen in figure 7. A plot of the error for sequence #4 using SMAT (original clustering) and R-SMAT (with *leaderP* clustering) is shown in figure 8. Both use a pre-learned shape model and robust fitting. The error plot of STASM is also shown. As it can be seen, R-SMAT shows lower error levels than the original SMAT and similar fitting error levels to STASM, with better robustness to self-occlusions.

6.4. Timings

Table 3 presents the average execution speed for R-SMAT in frames per second, for some representative con-



(a) *Type A* (b) *Type B*

Figure 6: Comparison of the performance of STASM and R-SMAT

figurations. The worst frame times are close to the limit, but these are infrequent cases. The original SMAT shows good average performance, but with a much higher variance. Processing times of STASM are also shown. As can be seen, STASM is much slower than the other methods.

Table 3: Execution time of different configurations of R-SMAT in frames per second

Configuration	Mean	Sdv	Worst frame
leader, 20×20	89.39	31.65	31.05
leaderP, 15×15	112.56	32.80	36.86
leaderP, 20×20	78.09	30.77	32.78
hierarchical, 15×15	102.60	24.15	48.57
SMAT, 15×15	85.98	48.22	27.25
STASM	2.17	0.13	1.96

Tests were run on a Xeon 2.2 GHz, running GNU/Linux, with GCC 4.2 as compiler. Multi-threading was not used and compiler optimizations were disabled (`-O0`).

7. Conclusions and future work

This paper has presented a face tracking method based on automatic appearance modeling.

The proposed face tracking method is an extension to Simultaneous Modeling and Tracking (SMAT). The incremental shape model of SMAT is replaced with a shape model built offline from handmarked data, and Huber function used for fitting. Three alternative clustering methods have been presented to model the appearance online. The first two are the *leader* algorithm and a modification of it, while the third one is a hierarchical method. The performance of R-SMAT and the original SMAT have been evaluated on the sequences of the RS-DMV dataset. The results show that the three alternative clustering methods improve the error levels of the original proposal. R-SMAT is able to process more than 100 frames per second, and obtains similar accuracy to STASM. The source code of R-SMAT is available under a free licence from the authors.

Future work will explore ways to make R-SMAT fully autonomous, improving the incremental shape model by applying Procrustes method in the shape clustering. Appearance clustering would benefit from better techniques to remove outliers from the model. Including a multi-scale approach to appearance modeling and model fitting would be of help in other scenarios where the size of the face changes noticeably. The R-SMAT has been used to track and model faces in this paper, but can be used for other deformable objects. The RS-DMV dataset will be extended with more sequences, more drivers and more diverse scenarios. Finally, the R-SMAT is to be made part of a driver monitoring system.

Acknowledgments

The authors would like to thank Ivan García and Noelia Hernández of RobeSafe for their work in recording the sequences, and Sebastian Bronte for his help in marking the images, as well as the drivers that participated. The outdoor recordings were made under project MOVI^{2}CON (`TRA2005-08529-C02-02`) and the simulator recordings under CABINTEC project (`PSE-370100-2007-2`). This work was supported in part by the Spanish Ministry of Science and Innovation under DRIVER-ALERT Project (`TRA2008-03600`), and Comunidad de Madrid under project RoboCity2030 (`S-0505/CPI/000176`). J. Nuevo was working under a researcher training grant from the Education Department of the Comunidad de Madrid and the European Social Fund.

References

Asharaf, S., Murty, M., 2003. An adaptive rough fuzzy single pass algorithm for clustering large data sets. Pattern Recognition 36 (12), 3015–3018.

(a) Seq. #1, Frame 1170 (b) Seq. #2, frame 2550 (c) Seq. #5, frame 50

(d) Seq. #5, frame 290 (e) Seq. #6, frame 1306 (f) Seq. #7, frame 1290

(g) Seq. #7, frame 2590 (h) Seq. #9, frame 2140 (i) Seq. #9, frame 3328

Figure 7: Examples from sequences with R-SMAT fitted

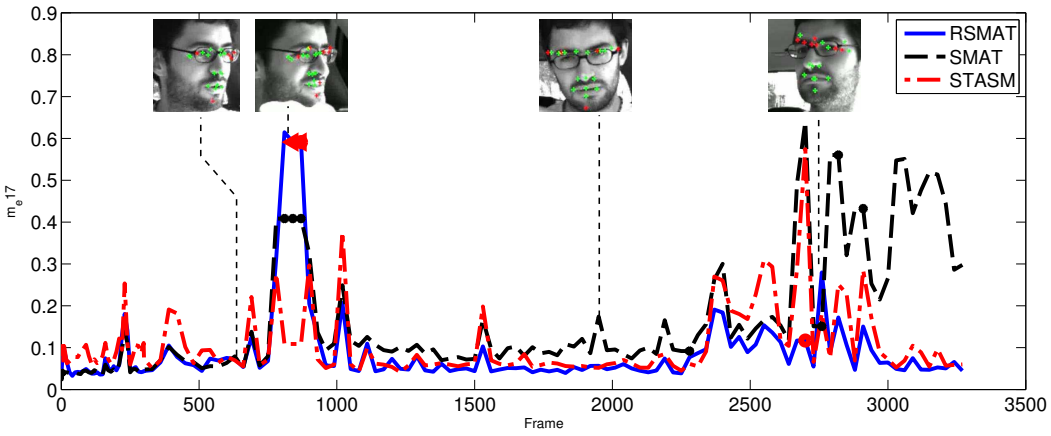


Figure 8: Error plots for SMAT, R-SMAT and STASM in sequence #4

Asharaf, S., Narasimha Murty, M., Shevade, S., 2006. Rough set based incremental clustering of interval data. Pattern Recognition Letters 27 (6), 515–519.

Avidan, S., 2007. Ensemble Tracking. IEEE Trans. Pattern Anal. Mach. Intell., 261–271.

Baker, S., Matthews, I., 2001. Equivalence and efficiency of image alignment algorithms. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition. Vol. 1. pp. 1090–1097.

Belhumeur, P., Hespanha, J., Kriegman, D., July 1997. Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. IEEE Trans. Pattern Anal. Mach. Intell. 19 (7), 711–720, special Issue on Face Recognition.

Bergasa, L. M., Nuevo, J., Sotelo, M. A., Barea, R., López, E., Mar. 2006. Real-time system for monitoring driver vigilance. IEEE Trans. Intell. Transp. Syst. 7 (1), 1524–1538.

Buenaposada, J. M., Muñoz, E., Baumela, L., 2006. Efficiently estimating facial expression and illumination in appearance-based tracking. In: Proc. British Machine Vision Conference. Vol. 1. pp. 57–66.

Buenaposada, J. M., Muñoz, E., Baumela, L., 2008. Recognising facial expressions in video sequences. Pattern Analysis and Applications 11 (1), 101–116.

Can, F., 1993. Incremental clustering for dynamic information processing. ACM Transactions on Information Systems (TOIS) 11 (2), 143–164.

Collins, R., Liu, Y., Leordeanu, M., 2005. Online Selection of Discriminative Tracking Features. IEEE Trans. Pattern Anal. Mach. Intell., 1631–1643.

Cootes, T., Jul. 2005. Timeline of developments in algorithms for finding correspondences across sets of shapes and images. Tech. rep., University of Manchester.

Cootes, T., Taylor, C., Cooper, D., Graham, J., 1995. Active Shape Models-Their Training and Application. Computer Vision and Image Understanding 61 (1), 38–59.

Cootes, T. F., Edwards, G. J., Taylor, C. J., Jan. 2001. Active appearance models. IEEE Trans. Pattern Anal. Mach. Intell. 23, 681–685.

Cristinacce, D., Cootes, T., 2006. Feature Detection and Tracking with Constrained Local Models. In: 17th British Machine Vision Conference. pp. 929–938.

Cristinacce, D., Cootes, T., 2008. Facial motion analysis using clustered shortest path tree registration. In: Proc. of the 1st Int. Workshop on Machine Learning for Vision-based Motion Analysis with ECCV. pp. 1–12.

Dowson, N., Bowden, R., 2005. Simultaneous modeling and tracking (SMAT) of feature sets. In: IEEE Conference on Computer Vision and Pattern Recognition 2005. Vol. 2. pp. 99–105.

Dowson, N., Bowden, R., 2006. N-tier simultaneous modelling and tracking for arbitrary warps. In: Proc. of the 17th British Machine Vision Conference. British Machine Vision Association. Vol. 1. p. 6.

Dryden, I., Mardia, K., 1998. Statistical Shape Analysis. John Wiley & Sons.

Ellis, L., Dowson, N., Matas, J., Bowden, R., 2007. Linear predictors for fast simultaneous modeling and tracking. In: Proceedings of 11th IEEE International Conference on Computer Vision, workshop on Non-rigid registration and tracking through learning, Rio de Janeiro, Brazil, IEEE computer society. pp. 1–8.

Gower, J., 1975. Generalized procrustes analysis. Psychometrika 40 (1), 33–51.

Grabner, H., Bischof, H., 2006. On-line boosting and vision. In: Proc. CVPR. Vol. 1. pp. 260–267.

Hager, G. D., Belhumeur, P. N., 1998. Efficient region tracking with parametric models of geometry and illumination. IEEE Trans. Pattern Anal. Mach. Intell. 20 (10), 1025–1039.

Hartigan, J., 1975. Clustering Algorithms. John Wiley & Sons, Inc. New York, NY, USA.

Huber, P. J., 1981. Robust Statistics. Wiley Series in Probability and Mathematical Statistics. Wiley-Interscience.

Jain, A., Murty, M., Flynn, P., 1999. Data Clustering: A Review. ACM Computing Surveys 31 (3).

Jepson, A., Fleet, D., El-Maraghi, T., 2003. Robust Online Appearance Models for Visual Tracking. IEEE Trans. Pattern Anal. Mach. Intell., 1296–1311.

Jimenez, P., Nuevo, J., Bergasa, L. M., 2009. Face Tracking and Pose Estimation with Automatic 3D Model Construction. IET Computer Vision.

Jurie, F., Dhome, M., 2002. Hyperplane approximation for template matching. IEEE Trans. Pattern Anal. Mach. Intell., 996–1000.

Kaneko, T., Hori, O., 2002. Template update criterion for template matching of image sequences. In: International Conference on Pattern Recognition. Vol. 2. pp. 1–5.

Kutila, M., 2006. Methods for Machine Vision Based Driver Monitoring Applications. Ph.D. thesis, VTT Technical Research Centre of Finland.

Lucas, B., Kanade, T., 1981. An iterative image registration technique with an application to stereo vision. In: International Joint Conference on Artificial Intelligence. Vol. 3. pp. 674–679.

Manning, C. D., Raghavan, P., Schütze, H., 2008. Introduction to Information Retrieval. Cambridge University Press.

Matsumoto, Y., Zelinsky, A., mar 2000. An algorithm for real-time stereo vision implementation of head pose and gaze direction measurements. In: Procs. IEEE 4th Int. Conf. Face and Gesture Recognition. pp. 499–505.

Matthews, I., Ishikawa, T., Baker, S., 2004. The Template Update Problem. IEEE Trans. Pattern Anal. Mach. Intell., 810–815.

Milborrow, S., Nicolls, F., 2008. Locating facial features with an extended active shape model. In: ECCV. pp. 504–513, `http://www.milbo.users.sonic.net/stasm`.

Pham, M., Cham, T., 2007. Online Learning Asymmetric Boosted Classifiers for Object Detection. In: Computer Vision and Pattern Recognition CVPR. pp. 1–8.

Pilet, J., Lepetit, V., Fua, P., June 2005. Real-time non-rigid surface detection. In: IEEE Conference on Computer Vision and Pattern Recognition 2007. San Diego, CA, pp. 822–828.

Rasmussen, E., 1992. Clustering algorithms. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, pp. 419–442.

Rogers, M., Graham, J., 2002. Robust active shape model search. In: In Proceedings of the European Conference on Computer Vision. Springer, pp. 517–530.

Seeing Machines, Aug. 2004. Facelab. URL `http://www.seeingmachines.com/facelab`

Seeing Machines, Aug. 2007. Driver state sensor. Http://www.seeingmachines.com/dss.html.

Segvic, S., Remazeilles, A., Chaumette, F., May 2006. Enhancing the point feature tracker by adaptive modelling of the feature support. In: European Conf. on Computer Vision, ECCV'2006. Vol. 3952 of Lecture Notes in Computer Science. Graz, Austria, pp. 112–124.

SmartEyeAG, 2009. AntiSleep. `www.smarteye.se`.

Spath, H., Bull, V., 1980. Cluster analysis algorithms for data reduction and classification of objects. Ellis Horwood.

Tong, Y., Liu, X., Wheeler, F., Tu, P., 2009. Automatic facial landmark labeling with minimal supervision. Computer Vision and Pattern Recognition, IEEE Computer Society Conference on 0, 2097–2104.

Victor, T., Blomberg, O., Zelinsky, A., aug 2001. Automating the measurement of driver visual behaviours using passive stereo vision. In: Procs. Int. Conf. Series Vision in Vehicles VIV9. Brisbane, Australia.

Vijaya, P. A., Murty, M. N., Subramanian, D. K., 2004. Leaders-subleaders: An efficient hierarchical clustering algorithm for large data sets. Pattern Recognition Letters 25 (4), 505 – 513.

Wang, Y., Lucey, S., Cohn, J., 2008. Enforcing convexity for improved alignment with constrained local models. In: IEEE International Conference on Computer Vision and Pattern Recognition (CVPR). pp. 1–8.

Yang, M., Kriegman, D., Ahuja, N., 2002. Detecting faces in images: a survey. IEEE Trans. Pattern Anal. Mach. Intell. 24 (1), 34–58.

Yin, Z., Collins, R., 2007. On-the-fly Object Modeling while Tracking. In: Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on. pp. 1–8.