Regressive Tree Structured Model for Facial Landmark Localization

Gee-Sern (Jison) Hsu, Kai-Hsiang Chang, Shih-Chieh Huang
Artificial Vision Lab., Dept Mechanical Engineering
National Taiwan University of Science and Technology
jison@mail.ntust.edu.tw

Abstract

Although the Tree Structured Model (TSM) is proven effective for solving face detection, pose estimation and landmark localization in an unified model, its sluggish runtime makes it unfavorable in practical applications, especially when dealing with cases of multiple faces. We propose the Regressive Tree Structure Model (RTSM) to improve the run-time speed and localization accuracy. The RTSM is composed of two component TSMs, the coarse TSM (c-TSM) and the refined TSM (r-TSM), and a Bilateral Support Vector Regressor (BSVR). The c-TSM is built on the low-resolution octaves of samples so that it provides coarse but fast face detection. The r-TSM is built on the mid-resolution octaves so that it can locate the landmarks on the face candidates given by the c-TSM and improve precision. The r-TSM based landmarks are used in the forward BSVR as references to locate the dense set of landmarks, which are then used in the backward BSVR to relocate the landmarks with large localization errors. The forward and backward regression goes on iteratively until convergence. The performance of the RTSM is validated on three benchmark databases, the Multi-PIE, LFPW and AFW, and compared with the latest TSM to demonstrate its efficacy.

1. Introduction

Face detection with landmark localization is a challenging problem because the face can be arbitrary in pose, expression, resolution and illumination condition. The solution generally consists of two steps, initialization and fitting. The former handles face detection and landmark initial localization, and the latter searches for the best located landmarks that minimize the model-based fitting error. One of the most promising models is the Deformable Part Models (DPMs), and the Constrained Local Models (CLMs) [2, 5, 1, 11] and Tree Structured Models (TSMs) [12, 3, 6] are among the most successful approaches in the DPM family. A CLM with joint shape and texture appearance is proposed in [2] to generate patch template detectors. The model is fitted to a facial image in an iterative manner by generating templates using the joint model and the current parameter estimates, correlating the templates with the target image to generate response images and optimizing the shape parameters so as to maximize the sum of responses. In [5], a fitting scheme, coined the Regularized Landmark Mean-Shift (RLMS), is proposed where the nonparametric likelihoods of landmark locations are maximized within a hierarchy of smoothed estimates. Because the discriminative regression-based fitting approaches have not received much attention in the CLM framework, the Discriminative Response Map Fitting (DRMF) is proposed in [1] and proven better than the RLMS in performance. Although the advancement through the aforementioned and other CLMs yields fast and accurate landmark fitting, an important issue is the initialization, i.e., initial face detection and landmark localization regardless of pose, illumination and other variables. It is noteworthy that quite a few CLMs use TSM for initialization, such as those in [1, 11], because TSM outperforms many detectors handling large rotations and unbalanced illumination [12, 4].

Unlike most CLMs that concentrate on the landmark fitting accuracy, the TSM can solve three tasks, namely initial detection, landmark localization and pose estimation in an unified framework [12]. However, the major disadvantage of the TSM is the heavy computation required at run time, substantially impeding its capability handling practical applications. We propose the Regressive Tree Structured Model (RTSM) for solving this speed issue and further improving accuracy. The RTSM is composed of two component TSMs, the coarse TSM (c-TSM) and the refined TSM (r-TSM), and a Bilateral Support Vector Regressor (BSVR). The c-TSM is built on the low-resolution octaves of samples so that it provides coarse but fast face detection. The r-TSM is built on the mid-resolution octaves so that it can locate the landmarks on the face candidates given by the c-TSM and improve precision. The r-TSM based landmarks are then processed using BSVR with shape traits to improve the overall localization accuracy.

The rest of the paper is organized as follows: a brief re-

view on the TSM is given in Sec.2. The proposed RTSM is presented in Sec.3, followed by the experiments on three benchmark databases reported in Sec.4. A conclusion of this study is given in Sec.5

2. A Review on Tree Structured Model

A tree structured model T consists of two components, V and E, where V is the set of parts, E is the geometrical connection of the parts. The former characterizes the features of a specific set of image patches and the latter configures how these patches are connected. The model with n parts can be defined in a feature pyramid by a $(n+2)$-tuple $(F_0, P_1, ..., P_n, \beta)$, where F_0 is the root filter, P_i is the part model for Part-i and β is a bias term. P_i is characterized by a 3-tuple $(F_i, s_i, d_{i,j})$, where F_i is the filter for Part-i, $s_i \in R^2$ specifies the location of Part-i, and $d_{i,j} \in R^4$ specifies the coefficients of a quadratic function that defines the deformation cost for each possible placement of Part-i relative to its parent Part-j [9, 6, 12]. Given an image I, the model can be applied to compute the following score $S(I, \mathbf{p})$ for a candidate facial region in the pyramid of I,

$$S(I, \mathbf{p}) = \sum_{i=0}^{n} F_i \cdot \phi(p_i) - \sum_{i,j \in E} d_{i,j} \cdot \rho_d(p_i, p_j) + \beta \quad (1)$$

where $\mathbf{p} = [p_0, ..., p_n]$, $p_i = [x_i, y_i]$ specifies the location of candidate Part-i on I, $\phi(p_i)$ is the patch feature computed at p_i; where $dx_{i,j} = x_j - x_i$ and $dy_{i,j} = y_j - y_i$. $\rho_d(p_i, p_j)$ is the shape deformation between p_i and p_j, which in [12] is computed as $\rho_d(p_i, p_j) = [dx_{i,j}\ dx_{i,j}^2\ dy_{i,j}\ dy_{i,j}^2]^T$, where $dx_{i,j} = x_j - x_i$ and $dy_{i,j} = y_j - y_i$.

When searching for the target object, we maximize (1) over all possible $\mathbf{p}$ so that the one with the most appropriate configuration $\mathbf{p}^*$ receives the highest score $S(I, \mathbf{p}^*)$. Because of the tree structure, the maximization of $S(I, \mathbf{p})$ can be performed via dynamic programming, which computes the highest score that Part-i passes to its parent Part-j as follows [9]:

$$n_i(p_j) = \max_{p_i} \left(g_i(p_i) + d_{i,j} \cdot \rho_d(p_i, p_j) \right) \quad (2)$$

$$g_i(p_i) = F_i \cdot \phi(p_i) + \sum_{k \in K(i)} n_k(p_i) \quad (3)$$

where $K(i)$ is the set of children of Part-i. (2) computes the highest scoring location of its child Part-i for every location of Part-j. (3) computes the local score of Part-i, at all pixel locations p_i, by collecting messages from $K(i)$. When scores are passed to the root part (i = 0), $g_0(p_0)$ gives the configuration with the best score for each root position. One can use these root scores to generate multiple detections in I by thresholding them and applying non-maximum suppression, and then backtrack to find the location and type of each part in each best-scored configuration by keeping track of the indices with score maxima.

The above algorithm is applied in [12] for face and landmark detection with the histogram of oriented gradients (HoG) used as the part feature $\phi(p_i)$ at location p_i. Because the scoring function (1) is linear in the part filters F_k, the spring parameters $d_{i,j}$ and bias β, it can be written as

$$S(I, \mathbf{p}) = \mathbf{q} \cdot \Phi(I, \mathbf{p}) \quad (4)$$

where

$$\mathbf{q} = [F_0, ..., F_n, d_{(k_0,l_0)}, ..., d_{(k_\gamma,l_\gamma)}, \beta] \quad (5)$$

$$\Phi(I, \mathbf{p}) = [\phi(p_0), ..., \phi(p_K), \rho_d(p_{k_0}, p_{l_0}), ..., \rho_d(p_{k_\gamma}, p_{l_\gamma}), 1]^T \quad (6)$$

where $(k_n, l_n) \in E$ denotes a parent-child pair and there are γ pairs in the model. $\Phi(I, \mathbf{c})$ can be a sparse vector when considering a mixture model with multiple components [6]. Given the model (4) and a training set with positive and negative samples, one can build a classifier using the latent support vector machine (LSVM), which in the training phase takes the following form:

$$\mathbf{q}^*, \zeta_n^* = \arg\max_{\mathbf{q}, \zeta_n} \left(\frac{1}{2} \mathbf{q} \cdot \mathbf{q}^T + c_a \sum_n \zeta_n \right) \quad (7)$$

$$\text{subject to} \quad \mathbf{q} \cdot \Phi(I, \mathbf{p}^+) \geq 1 - \zeta_n \quad \forall n \in S^+$$
$$\mathbf{q} \cdot \Phi(I, \mathbf{p}) \leq -1 + \zeta_n \quad \forall n \in S^-, \ \forall \mathbf{p}$$
$$q_k \leq 0, \quad \forall k \in K_a$$

where $\zeta_n > 0$ is an empirical measure of the misclassification error, $\sum_n \zeta_n$ gives an upper bound on the training error, c_a is a parameter that controls the trade-off between the class margin and error, $\{q_k\}$ is a subset of $\mathbf{q}$ and K_a are the indices of the quadratic spring terms in $\mathbf{q}$.

Because of heavy computation required at run-time detection, the TSM in [12] can hardly handle practical applications. To speed up, the authors proposed "part sharing". Instead of using a mixture model for each part at each pose, the same parts across a range of poses are aggregated and modeled by a mixture. Two extreme cases are considered, one with each part at each pose modeled by a mixture, and the other with the same part across all poses modeled by a mixture. The former results in Model p-1050 that accounts for 1050 independent parts, and the latter gives Model p-99 that accounts for 99 share parts. An intermediate model, p-146, is also considered and compared with the two extremes. It is shown in [12] that p-1050 yields the most accurate landmark localization and pose estimation; however, the part-sharing models, p-146 and p-99, come with much faster run-time detection (p-99 is almost $10\times$ faster than p-1050, which is around 40 secs per image) on the price of performance degradation. The landmark localization error in p-99 is $\approx 13\%$ more than that of p-1050.

3. Regressive Tree Structured Models

The Regressive Tree Structured Model (RTSM) aims at handling the issues of sluggish run time and poor detection rate on small faces (40×40 or less) using the TSM in [12]. The training samples considered in [12] are around 200×200 pixels. Faces of such a size allow a large patch at each landmark, e.g., 20×20 in [12], and such a large patch can confine local characteristics good as the part feature $\phi(I, p_i)$ in the training phase. However, when applying such a model to detect small faces and localize landmarks, either the faces are hardly detected or the landmarks are poorly located. This is because the part patches lose sufficient supports and can hardly be confined accurately on such small faces. When tested on the CMU-MIT database [10], in which many faces are of 40×40 or less, p-1050 gives detection rate 29% while the Viola-Jones detector offered in OpenCV gives 72%. Similar difficulty is also observed in manual annotation. For the choice of 68 landmarks on the whole face with 20 of them on the mouth, one would feel much harder when annotating these many landmarks on a 40×40 face than on a 200×200 face, as the landmarks in the former case are too close to each other and the associated patches are too small to confine sufficient characteristics. However, it can be feasible on a 40×40 again if only a small fraction, e.g., 10 of the landmarks are to be annotated because they can be made apart in such a case and each can be with a sufficiently large patch. This observation inspires the development of RTSM, which considers different numbers of landmarks in different scales and uses a partial set of landmarks to estimate the dense set of landmarks. Although multi-resolution has been considered in the Part-based Model (PBM) [6] and TSM, it is used in a completely different way in our approach.

The RTSM is composed of a coarse TSM (c-TSM), a refined TSM (r-TSM) and an BSVR (Bilateral Support Vector Regressor) . The c-TSM is designed for fast detection of face candidates which are further processed by the r-TSM for locating landmarks and pose estimation. Because only a small number of parts are considered, the c-TSM's training and run-time detection can be orders of magnitude faster than the original TSM in [12], on the cost of less accurate landmark localization and pose estimation. The landmarks considered in the r-TSM are a partial set of those in the original TSM, and therefore the training and run-time detection of the r-TSM is also much faster than of the original TSM. Given the r-TSM landmarks, the rest of the dense set of landmarks are estimated using the forward BSVR instead of the time-consuming part-based model. The BSVR is trained on the shape model with dense landmarks only and without considering appearance features, resulting in a fast landmark detector. The landmarks detected by the forward BSVR can be used in the backward BSVR to further improve the landmark localization accuracy. The details of the RTSM can be split into the following steps:

1. Assuming that the faces considered in the original TSM are of size z_0, the RTSM considers the octaves down to the second order, namely z_1 and z_2 with scale factors $1/2$ and $1/4$, respectively. The n_0 landmarks on the z_0-scaled faces are assumed to be kept well with sufficiently large patches (of size s_1) on the downsized octave z_1, but octave z_2 is too low in resolution to preserve the n_0 landmarks and better characterized by a different set of landmarks. The c-TSM is built on octave z_2 with n_2 landmarks selected from the n_0, and each landmark is with a patch of size s_2. As c-TSM is built on low resolution faces, it aims at fast detection of faces with a coarse estimate to their poses rather than precise landmark localization.

2. Given a face detected by the c-TSM, the r-TSM relocates a different set of landmarks for identifying the pose of the face. The r-TSM is built on z_1-scaled faces with n_1 landmarks selected from the original n_0. The n_1 landmarks are empirically determined so that the poses can be accurately identified and the rest $n_0 - n_1$ landmarks can be better located in the shape-based regression phase.

3. Given the r-TSM located n_1 landmarks, the forward BSVR estimates the rest $n_{0,1}$ landmarks, where $n_{0,1} = n_0 - n_1$. Let $\mathbf{y} = [y_1, y_2, \cdots, y_{n_{0,1}}]$ denote the forward BSVR estimated landmarks and $\mathbf{x} = [x_1, x_2, \cdots, x_{n_1}]$ is the set of n_1 r-TSM located landmarks, then $y_j = f_j(\mathbf{x})$, $j = 1, \cdots, n_{0,1}$ where the BSVR $f_j(\mathbf{x})$ is in the following form:

$$f_j(\mathbf{x}) = \sum_{i=1}^{l_j} \alpha_{i,j} k_j(\mathbf{x}_{i,j}, \mathbf{x}) + b_j \tag{8}$$

where $k_j(\cdot)$ is the forward BSVR kernel, $\{\alpha_{i,j}\}$ are the coefficients, $\{\mathbf{x}_{i,j}\}$ are the support vectors and b_j is the bias of f_j for estimating y_j. These parameters are determined on the training set in the training phase. For poses $> 75^o$ with occlusion, such as the case shown in Fig.1, the landmarks on the occluded region can be better located using only the r-TSM landmarks along the facial profile.

4. Based on the comparison with the ground truth in the training set, the landmarks located by the r-TSM and forward BSVR reveal different localization accuracy one another. We found the landmarks located at certain locations are more accurately located than those on other locations. Those with low localization errors are hence selected as references to train the backward BSVR to relocate those with large localization errors. At run time, the former are used as the input to the

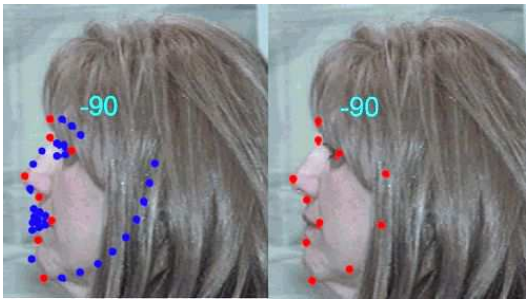

Figure 1. BSVR can effectively handle the cases with parts completely covered by different textures. The right one shows the overall r-TSM located landmarks, and the left one shows the landmarks relocated by the BSVR (in blue) using the r-TSM landmarks on the profile only as reference (in red)

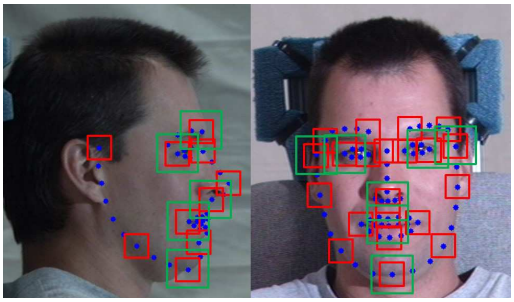

Figure 2. The blue dots enclosed by green bounding boxes are c-TSM landmarks, those in red bounding boxes are r-TSM landmarks and the rest are landmarks located by forward BSVR

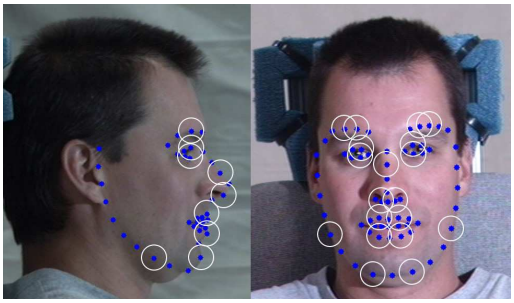

Figure 3. Blue dots are the landmarks obtained by the r-TSM and forward BSVR, and those enclosed by white circles are selected as reference points for the backward BSVR to relocate the other landmarks

backward BSVR to relocate the rest of the dense landmarks. Fig. 3 shows the landmarks selected as the input to the backward BSVR.

The novelty of the RTSM can be addressed as follows:

1. The RTSM splits the model into a coarse level for holistic object (face) detection and a refined level for component localization. The holistic coarse search scans the whole given image in high speed for locating the target candidates, followed by the refined component search which is only performed on the target candidates. Therefore, RTSM can be orders of magnitude faster than the TSM.

2. Unlike the TSM in which all parts are of the same size, the parts in the RTSM can vary in size as the landmarks are defined in different octaves with patches of different sizes, as shown in Fig. 2, offering more degrees of freedom to model faces. Experiments show that the landmark patches in the c-TSM are better made larger than those in the r-TSM in terms of the ratio to the whole face, as the former require larger ratio of regions to better confine sufficient characteristics.

3. The inclusion of BSVR further improves the robustness against occlusions, and substantially reduce the computational complexity. Although the TSM can handle the cases with one or a few parts partially occluded, it often fails in the cases with parts completely covered by different textures as the sample shown in Fig.1. The BSVR can effectively and efficiently handle such cases using the non-occluded landmarks in the r-TSM as reference points. The case on the right of Fig.1 shows that the landmarks between the chin and ear are occluded and poorly located using r-TSM, which computes local features for matching. However, such mistakes can be corrected if using the r-TSM landmarks on the profile only as the input to the BSVR, which considers shape model only and leaves along local features, as shown in the case on the left. As the pose is estimated by the r-TSM at run time, the landmarks on the profile are also located from the estimation and can be used as references.

4. Experimental Evaluation

The experiments were designed to compare the performance of the proposed RTSM and the TSM in [12], which was proven state of the art for handling face detection, pose estimation and landmark localization in an unified model. Three benchmark databases were used in the experiments, Multi-PIE [8], LFPW [7] and AFW [12]. We followed similar settings as in [12] with 1100 randomly selected faces that cover all 13 viewpoints with normal and 5 arbitrarily selected illumination conditions and 5 expressions under normal illumination condition from the Multi-PIE, and 1600 non-facial images for training. A disjoint set of 1500 faces was selected for testing. This Multi-PIE trained model was also tested on the LFPW and AFW databases. Each face in the training set was around 200×200 pixels, considered as size z_0 in our test, and the TSMs in [12] were built on faces this size. The c-TSM was built on the octave z_2 with scale factor 1/4, i.e., faces in 50×50 or so; and r-TSM was on z_1 with scale factor 1/2, i.e., around 100×100. Testing across various settings for the image patches for computing the

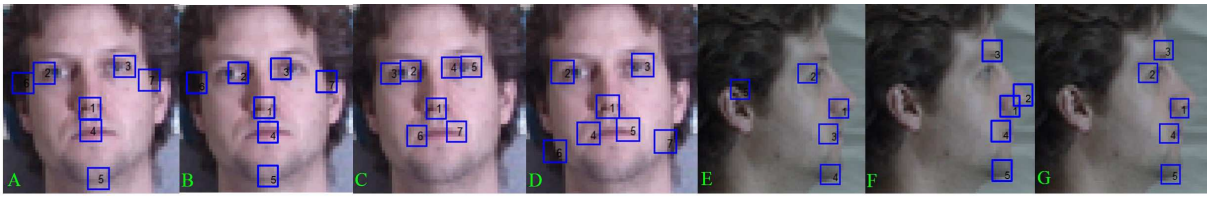

Figure 4. Payouts of landmarks considered in the determination of settings for c-TSM

Figure 5. The bottom row shows cases processed by c-TSM(7/5) with false positives, which are blocked away after combining r-TSM(21/11), as shown in the top

HOG features, for compromising between speed and precision each part in the c-TSM was chosen as 6×6 pixels with 3-by-3 cells in it and each cell was 4×4 pixels overlapped with its neighboring cell for one row and one column. Each part in the r-TSM was chosen as 9×9 pixels with 3-by-3 cells and each cell was 3×3 pixels without overlap. The rest of HOG settings were the same as those in [12]. All tests were run on a Windows-7 PC with i7 (3.4GHz) processor and RAM 16GB.

Before comparing the performance of RTSM and TSM, tests were carried out on the training set for the determination of the best settings on the RTSM model parameters, including the landmarks appropriate for defining the c-TSM and r-TSM. The c-TSM aims at fast detection of faces with a small number of parts trained on low resolution data. When the parts increase, the detection rate increases, on the price of longer processing time (to be shown in the experimental result). An appropriate number of parts balance between high detection rate and short processing time. Besides, the locations of landmarks also affect the detection rate, and those on the r-TSM affect both the pose estimation and the BSVR based landmark localization. Comparing the performances with different numbers of landmarks on the training set, the c-TSM was chosen with 7 landmarks for yaw angles between $\pm 45^o$ and 5 landmarks for yaw beyond that range,

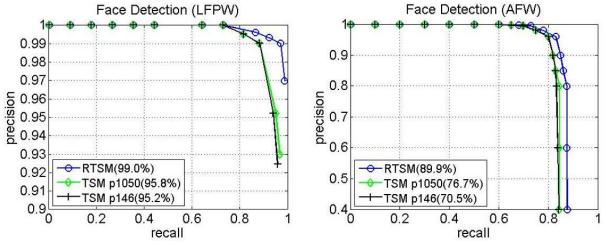

Figure 6. Precision-recall rates for LFPW and AFW, the percentages are average precision (AP)

and the r-TSM was chosen with 21 landmarks for yaw between $\pm 45^o$ and 11 landmarks for yaw beyond that range. Fig.4 shows different c-TSM landmark locations compared in the experiments, and the configurations in A and G are chosen for yaw between $\pm 45^o$ and the angles beyond the range, respectively. The last selected configuration, shown in Fig.2, consists of landmarks located by the c-TSM, r-TSM, forward BSVR and backward BSVR. Note that the c-TSM landmarks are used only for face detection, and the landmark localization solely depends on the r-TSM and forward/backward BSVRs. To summarize the parameter settings considered, Table 1 gives the range of each parameter tested and compared in our experiments.

The face detection performance was measured by the

Table 1. The RTSM parameters tested/compared in the experiments

	Part Num. (Frontal/Profile)	Part Size	Cell Num.	Cell Size
c-TSM	7/5, 10/8, 15/10	4x4, 5x5, ..., 9x9	2x2, 3x3	2x2, 3x3, 4x4
r-TSM	15/10, 21/11, 36/24, 68/39	4x4, 6x6, 9x9, 12x12, 21x21	2x2, 3x3, 5x5, 7x7	2x2, 3x3, ..., 7x7

Table 2. Run-time speeds, face detection rates in AP (Average Precision), landmark localization (Lmk Loc. in percentage of landmarks with error $< 5\%$) and pose estimates (percentage of faces with $< 15^o$ error) of the c-TSM (7/5), c-TSM (15/10), r-TSM (21/11), r-TSM (35/20), cr-TSM (i.e., c-TSM+r-TSM) and RTSM compared with p-1050 and p-146 [12] on Multi-PIE

Model	TSM p-1050	TSM p-146	c-TSM (7/5)	c-TSM (15/10)	r-TSM (21/11)	r-TSM (35/20)	cr-TSM (7/5,21/11)	RTSM
Time/Face (s)	25	2.9	0.05	0.1	1.5	2.2	0.25	0.25
Face Det.(AP)	100	96.2	100	100	100	100	100	100
Lmk Loc.(%)	86.9	72.6	19.2	23.4	74.8	85.8	74.8	87.7
Pose Est.(%)	96.2	90.6	75.6	81.9	96.3	95.9	96.3	96.3

Table 3. Run-time speeds, face detection rates in AP (Average Precision) and landmark localization (Lmk Loc. in percentage of landmarks with error $< 5\%$) of the c-TSM (7/5), c-TSM (15/10), r-TSM (21/11), r-TSM (35/20), cr-TSM (i.e., c-TSM+r-TSM) and RTSM compared with p-1050 and p-146 [12] on LFPW (The ground truth of pose is unavailable for LFPW)

Model	TSM p-1050	TSM p-146	c-TSM (7/5)	c-TSM (15/10)	r-TSM (21/11)	r-TSM (35/20)	cr-TSM (7/5, 21/11)	RTSM
Time/Face (s)	26	3.8	0.06	0.1	1.7	2.6	0.3	0.31
Face Det.(AP)	95.8	95.2	98.7	99.1	99.0	99.6	99.0	99.0
Lmk Loc.(%)	61.4	42.5	6.7	12.0	58.2	60.2	58.2	71.5

Table 4. Run-time speeds, face detection rates in AP (Average Precision), landmark localization (Lmk Loc. in percentage of landmarks with error $< 5\%$) and pose estimates (percentage of faces with $< 15^o$ error) of the c-TSM (7/5), c-TSM (15/10), r-TSM (21/11), r-TSM (35/20), cr-TSM (i.e., c-TSM+r-TSM) and RTSM compared with p-1050 and p-146 [12] on AFW

Model	TSM p-1050	TSM p-146	c-TSM (7/5)	c-TSM (15/10)	r-TSM (21/11)	r-TSM (35/20)	cr-TSM (7/5, 21/11)	RTSM
Time/Face (s)	49	6.5	0.1	0.14	2.8	3.6	0.7	0.7
Face Det.(AP)	88.7	87.7	76.6	89.0	89.9	91.2	89.9	89.9
Lmk Loc.(%)	76.7	70.5	29.5	53.4	76.4	78.9	76.4	80.0
Pose Est.(%)	81.0	77.2	60.4	67.5	82.0	86.2	82.0	82.0

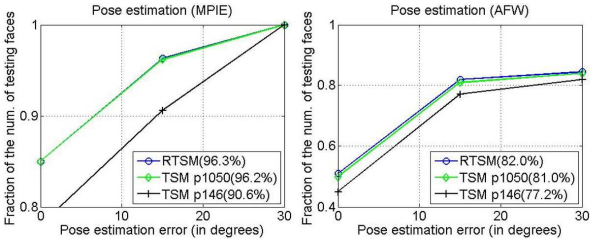

Figure 7. Pose est. errors and percentages with error $< 15^o$

precision-recall rate. The pose estimation was measured by the cumulative error from the ground truth in 15^o each unit. The landmark error was measured by the pixel distance normalized by the face size computed as the mean of its height plus width. The best two TSMs, namely p-1050 and p-146, proven to outperform other methods in [12], were selected to compare with the RTSM, and the codes were taken off the shelf from the link given in the paper. Although the DRMF [1] was claimed to outperform the TSMs, it was excluded in this comparison as it only worked for poses within $\pm 45^o$ and used TSM for initialization.

The most appealing finding of this study may be the run time speed, shown in Tables 2 to 4, for the tests on Multi-PIE, LFPW and AFW, respectively. The RTSM appears orders of magnitude faster than p-1050 while revealing better performances in all three tasks. To compare TSMs with different numbers of parts, c-TSM(7/5), c-TSM(15/10), r-TSM(21/11) and r-TSM(35/20) were built and studied, where c-TSM(7/5) refers to a c-TSM with 7 landmarks for yaw angles between $\pm 45^o$ and 5 landmarks for yaw beyond that range. As expected, the c-TSM is the fastest, on the price of inaccurate pose esti-

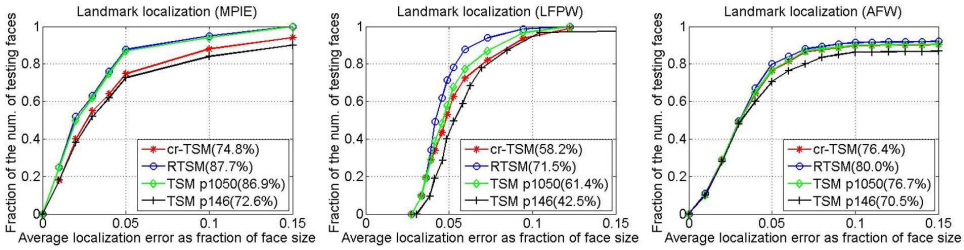

Figure 8. Landmark errors and percentages with errors $< 5\%$

mation and landmark localization. When the parts (landmarks) increase, accuracies improve with longer processing time. The cr-TSM(7/5,21/11) is the combination of the faster c-TSM(7/5) and r-TSM(21/11). cr-TSM(7/5,21/11) reveals comparable performances in face detection and pose estimation, but is inferior to p-1050 in landmark localization. Combining cr-TSM(7/5,21/11) and BSVR, the RTSM shows desired performances with a good balance between accuracy and speed. Fig.5 shows a few cases processed by the c-TSM(7/5) with quite a few false positives, which are blocked away after combining r-TSM(21/11).

Among the three databases, the Multi-PIE appears the least challenging as it is made in the lab, while the other two are made in the wild. Using the RTSM with cr-TSM(7/5,21/11), the precision-recall rates for LFPW and AFW are shown in Fig.6 with average precision (AP) given in the parentheses. The RTSM outperforms both p-1050 and p-146 as the RTSM performs better detecting small faces and the r-TSM effectively cuts down the false positives. Since LFPW does not offer ground truth in pose, it is excluded in the pose estimation comparison. The results on Multi-PIE and AFW are shown in Fig.7. The RTSM performs similarly well as p-1050 because both use similar TSMs but with different numbers of parts for pose estimation. The landmark localization on Multi-PIE, LFPW and AFW is given in Fig.8. To better observe the contribution of the BSVR, we added in the cr-TSM, which stands for c-TSM and r-TSM without the shape-based BSVR part. It shows that the BSVR improves the cr-TSM landmark localization and outperforms p-1050, which shows better performance than the cr-TSM alone.

5. Conclusion

The proposed RTSM takes the advantages of a coarse model for the global search followed by a refined model for the local search, and enhances localization accuracy using shape-based regression. It differs from the TSM in not just the architecture but also the additional means of relocating the landmarks with large localization errors. Validated on three benchmark databases, the RTSM reveals competitive performance and processing speed. As different parts contain different spatial frequencies and orientations, the next phase of this research focuses on models with parts of different sizes and orientations.

References

[1] S. C. A. Asthana, S. Zafeiriou and M. Pantic. Robust discriminative response map fitting with constrained local models. *Proc. IEEE Int. Conf. Computer Vision and Pattern Recognition (CVPR)*, Jun 2013. 1, 6

[2] D. Cristinacce and T. Cootes. Feature detection and tracking with constrained local models. In *British Machine Vision Conference (BMVC)*, 2006. 1

[3] P. F. Felzenszwalb and D. P. Huttenlocher. Pictorial structures for object recognition. *International Journal of Computer Vision (IJCV)*, 2005. 1

[4] G. S. Hsu and T. Y. Chu. A framework for making face detection benchmark databases. *IEEE Trans. on Circuits and Systems for Video Technology (TCSVT)*, vol.24(2):230–241, Feb 2014. 1

[5] S. L. J. M. Saragih and J. F. Cohn. Deformable model fitting by regularized landmark mean-shift. *International Journal of Computer Vision (IJCV)*, 91(2):200–215, Sep 2011. 1

[6] D. M. P. F. Felzenszwalb, R. B. Girshick and D. Ramanan. Object detection with discriminatively trained part based models. In *Pattern Analysis and Machine Intelligence (PAMI)*, 2010. 1, 2, 3

[7] D. J. K. P. N. Belhumeur, D. W. Jacobs and N. Kumar. Localizing parts of faces using a consensus of exemplars. In *Proc. IEEE Int. Conf. Computer Vision and Pattern Recognition (CVPR)*, pages 545–552, 2011. 4

[8] T. K. R. Gross, I. Matthew and S. Baker. Multi-pie. In *IEEE Int. Conf. Automatic Face and Gesture Recognition (FG)*, 2008. 4

[9] D. Ramanan. Part-based models for finding people and estimating their pose. *Visual Analysis of Humans (VAH)*, pages 199–223, 2011. 2

[10] H. Rowley, S. Baluja, and T. Kanade. Neural network-based face detection. *Pattern Analysis and Machine Intelligence, IEEE Trans.*, 20(1):23–38, 1998. 3

[11] G. Tzimiropoulos and M. Pantic. Gauss-newton deformable part models for face alignment in-the-wild. *Proc. IEEE Int. Conf. Computer Vision and Pattern Recognition (CVPR)*, Jun 2014. 1

[12] X. Zhu and D. Ramanan. Face detection, pose estimation, and landmark localization in the wild. In *Proc. IEEE Int. Conf. Computer Vision and Pattern Recognition (CVPR)*, pages 2879–2886, 2012. 1, 2, 3, 4, 5, 6