

# ESTIMATING FACIAL POSE FROM A SPARSE REPRESENTATION

*Hankyu Moon and Matt L. Miller*

NEC Laboratories America  
4 Independence Way  
Princeton, NJ 08648

## ABSTRACT

We present an approach to estimate the poses of human heads in natural scenes. The essential features for estimating the head pose are the positions of the prominent facial features relative to the position of the head. We have developed a high-dimensional, randomly sparse representation of a human face using a simplified facial feature model. The representation transforms a raw face image into a vector representing how well the image matches large number of randomly-posed and shaped head models. This transformation is designed to collect salient features of the face image that is useful to estimate the pose, while suppressing any irrelevant variations of face appearance. The relation between the sparse representation and the pose is learned using the SVR (Support Vector Regression). The sparse representation combined with SVR is shown to estimate the pose more quickly and accurately than SVR applied to raw images.

## 1. INTRODUCTION

One of the main hurdles in face recognition is obtaining robustness to variations in facial pose. Many existing face-recognition systems yield good results when comparing faces in frontal pose, but their performance drops off markedly as one of the faces moves toward profile [8]. Recently, however, some systems have been proposed which explicitly compensate for facial pose, and yield vastly better results [1, 2]. These systems require an estimate of the facial pose in each image as input, which motivates us to examine the problem of obtaining such an estimate automatically. Ultimately, when given an image of a single face, we will need to estimate all six pose parameters:  $x$  and  $y$  location, scale, yaw (rotation around the neck, from left profile to right profile), pitch (rotation up and down), and roll (rotation in the image plane). A seventh parameter, focal-length of the camera lense, may or may not be required. At present, however, we concentrate only on finding the two rotations that are out of the image plane – yaw and pitch – in images of faces that have been manually rotated to upright and scaled and translated to a canonical size and position. Although we present results on the PIE database [9] for comparison with other systems, our main emphasis is on unsystematic, natural images, with maximal variation in pose, expression, lighting, background, image quality, etc.

Quite a few methods for facial pose estimation have been proposed [3, 5, 4]. Of particular relevance to our new method are [5] and [4]. In [5], images are projected into a linear subspace obtained by applying PCA to images of faces in different poses. This reduces the dimensionality of the data while maintaining much of the image variation due to pose. In [4], Support-Vector Regression (SVR) was applied to map images into pose estimations.

The method we propose here is based on two observations. First, intuitively, facial pose can be estimated by looking at the locations of facial features in the image. Second, less intuitively, we can estimate the locations of those features by simply looking for them in a random collection of places. This latter observation comes from work done on face tracking using particle systems [6]. These observations lead us to analyze each image by correlating it with a set of feature detectors computed for a prespecified, but randomly-generated, set of facial shapes and poses. The resulting vector is a sparse representation of the face and its pose. We train an SVR system to map these vectors into yaw and pitch angles.

Thus, as in [5], we reduce the dimensionality of the problem with a linear projection. However, our projection is based on *a priori* knowledge of facial features, rather than statistical analysis of facial images. Like [4], we apply SVR to obtain our angle estimates. However we do not apply it to the raw image.

Our sparse representation of facial images is described in more detail in Section 2, and the training of our SVR is described in Section 3. This is followed, in Section 4, by a description of experiments we have performed on natural images and on the PIE database. The results of the experiments (Section 5) indicate that our sparse representation captures enough information to yield good pose estimates, and improves both performance and speed over application of SVR to raw pixels.

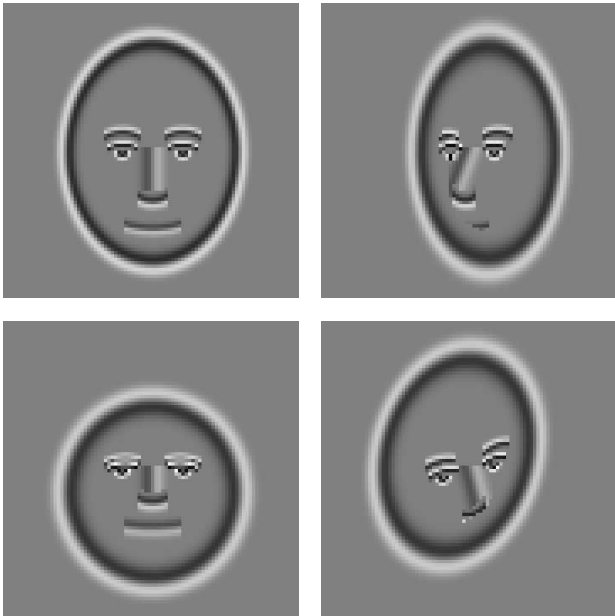
## 2. SPARSE REPRESENTATION OF FACIAL IMAGES

We have designed a sparse representation of human face, which captures the unique signatures of a human face effectively, while facilitating the estimation of the head position and pose. The representation is a collection of projections to a number of randomly generated possible configuration of the human face. Each projection corresponds to a pose of the head along with a configuration of its facial features. The projec-

tions should respond to changes in pose and feature configuration, while largely ignoring other image variations such as lighting, hairstyle, and background.

Our motivation for using a random, sparse representation of objects comes from the popular particle filtering approach to tracking and motion estimation problems [6]. Randomly generated motion parameter vectors, along with weights computed from image measurements, are known to efficiently estimate and propagate probability densities of the motion vectors, even when the state space is very high-dimensional.

We compare the features of our model with the image using the shape filters introduced in [7]. Basically, we use a simple parametric model of a head, in which a parameter vector determines the head shape and the locations of facial features upon it. For a given pose and parameter vector, we predict the locations of 10 straight and curved edge gradients in the image, and compute a specific filter for each. The correlation between the image and this filter gives a measure of how likely the image is to contain that edge, and therefore how likely it is to match a face with the given pose and model parameters. Some sets of 10 filters generated using the 3D model of the head and perspective projection are shown in Figure 1. Note that each line and curve in these images is a separate filter. The details of how these are computed can be found in [6].



**Fig. 1.** Combined feature detectors (filters) for head models in several poses.

In a particle-filtering approach, we would generate filters for a random collection of pose and parameter settings, and then compute a weighted average of those pose parameters, with weights based on the correlations between the image and the filters. Unfortunately, this approach will correctly estimate the pose of the head only if the random samples are generated

in a region that is close to the true state in the parameter space, which means we need an initial estimate of the pose. When such an initial estimate is not given, we should generate the random particles so that they span the wide range of parameter values to cover the correct value. Some of the filters, however, will pick up responses from irrelevant regions such as facial boundaries, or the hair, and give bias to the estimates.

Such off-match responses, however, also provide useful information about the pose of the head. For example, if the 'left eye filter' yields a very strong response when it is slightly moved (or 'rotated') to the right and keeps the level of response consistently when moved along the vertical direction, it is probable that the face is rotated to the left. This observation leads us to make use of the whole set of representations that cover a wide range of poses and model parameters. The face image will respond to the projections close to the true pose, and form a sharp peak, not necessarily a global maximum, around it. Other off-match projections could generate sufficient responses, yet the collective response will yield different shape.

### 3. GENERATION OF SAMPLES AND THE SUPPORT VECTOR REGRESSION

A large number of samples  $\{X^n | n = 1, 2, \dots, N\}$  that represent the pose of the model and the position and shapes of the facial features are generated. Each vector  $X_n$  then constructs the set of shape filters that will compute the image responses

$$R_n = \{eyel_n, eyer_n, bro_l_n, bro_r_n, iris_l_n, iris_r_n, nose_n, \text{noseprofile}_n, mouth_n, head_n\} \quad (1)$$

where each of  $eyel_n, eyer_n$ , etc. is a 2D pattern that matches one predicted edge gradient. There are 10 predicted edges, so we obtain  $10N$  filters. Note that a filter matched to the head boundary (to yield the response  $head_n$ ) is also used to compare the relative positions of the features to the head.

Computing the correlation between an image and all these  $10N$  filters is a linear transformation. We found, however, that the absolute values of these correlations produced better pose estimates.

Given a set of training images along with the pose:  $\{(I_m, \phi) | m = 1, 2, \dots, M\}$  ( $\phi$  can be  $\theta_x, \theta_y$ , or  $\theta_z$ ), we apply the above procedure to each image to generate the sparse representations  $\{X_m = (X_m^n) | n = 1, \dots, N | m = 1, 2, \dots, M\}$ . These linearly transformed image features are fed to the Support Vector Regression (SVR) algorithm to train the relation between  $X_m$  and  $\phi$ . SVR is a variant of Support Vector Machines [11], and known to approximate a high-dimensional functional relation effectively [10].

The regression problem is to find a functional relation  $f$  from the sparse representation to the pose angles:

$$f_\phi : X_m \mapsto g(\phi), \quad \phi = \theta_y \text{ or } \theta_x \quad (2)$$

where  $g$  is a nonlinear function of the angle  $\phi$ .

## 4. EXPERIMENTS

We trained and tested the proposed system on a large corpus of natural images. We also tested the system with the “illum” images in the PIE database, but none of the PIE images were used during training. For comparison, we conducted the same tests on a second SVR system working directly from the (histogram-equalized) raw face data, rather than from our sparse representation<sup>1</sup>.

All the faces were manually annotated with a simple tool for specifying facial location and approximate pose. The user interface for this tool is shown in Figure 2. To annotate a face, the user first clicks on the midpoint between the eyes and the center of the mouth. The tool then draws a perspective grid in front of the face, and the user moves the mouse to adjust the direction of this grid, until it appears as though it is parallel to the face plane. This process yields estimates for all six pose parameters.

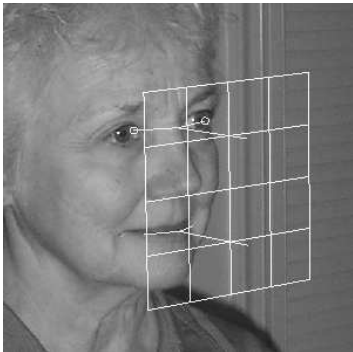


Fig. 2. Screenshot from annotation tool.

In this manner, we annotated roughly 30,000 faces in images from various sources. Each face was then cropped and scaled so that the eye midpoint and mouth midpoint appeared in canonical positions, 20 pixels apart, in an  $80 \times 80$ -pixel image. The resulting images were mirrored horizontally, to yield roughly 60,000 faces.

Next, we divided the images into two sets, one for training and the other for testing. Finally, we removed about two thirds of the faces from these sets to obtain a roughly uniform distribution of annotated yaw angles. Unfortunately, the amount of variation in pitch was not sufficient to do the same, so the distribution of pitch angles in our data is biased toward  $0^\circ$ . The final numbers of faces used for training and testing were 22,508 and 3,290, respectively.

We also manually annotated the PIE images, but we did not divide them into separate training and testing sets, nor did we discard any images to attain a uniform pose distribution. The

<sup>1</sup>Note that our second system is *not* an implementation of the system proposed in [4]. Rather, it is as similar to our first system as possible, without using the sparse representation. Thus, any difference in performance between the two indicates the effect of the sparse representation, but does not indicate any comparison with [4].

PIE images were used only for testing. In our experiments, we used  $N = 500$  pose/model-parameter combinations for computing the sparse representations, so our representations were 5000-dimensional vectors. Computing the representation of an image required about 4.5 million multiplications and additions.

## 5. RESULTS

The results of the training were that 35,151 support vectors were selected when using our sparse representations, while 35,993 support vectors were selected for raw pixel data. This means that applying SVR with sparse representations requires about 4.5 million multiplications and additions to compute the representation, plus 35,151 5000-dimensional distance calculations, for a total of about 180 million operations. By comparison, with raw pixel data, we require 35,993 6400-dimensional distance calculations, at a cost of about 230 million operations. Thus, applying SVR with raw pixels is roughly 28% slower than with our sparse representations.

Figure 3 shows a few examples of pose estimates obtained using our system. The left column of the figure shows original images from the PIE database. The right column shows synthetic images, generated by positioning a 3D face model with the poses obtained from our system. This figure illustrates that the system yields plausible pose estimates.

It is worth noting that training the SVR to produce the complex number  $g(\phi) = (\cos(\phi), \sin(\phi))$  of the angle results in a better performance than simply training it to output  $\phi$ . We suspect that this is because the relation from the pose angle to the image feature locations is essentially trigonometric.

Figure 4 is a graph showing the frequency with which poses are “correctly” estimated, assuming various levels of error tolerance in our eventual application. A pose is considered to have been correctly estimated if both the yaw and pitch estimate are within a given angle (shown on the x axis) of the manual annotation. The top line in the graph shows the results obtained using the sparse representations with the test set of natural images. The middle line shows results for the same system with the PIE database. The bottom line shows results obtained without the sparse representations.

Note that, as the annotations themselves are only approximate, a certain level of error is unavoidable, and we do not know how much error that is. Nevertheless, there are two significant conclusions that can be drawn from Figure 4.

First and foremost, the performance of SVR applied to our sparse representations is clearly superior to SVR applied to raw pixel data. With sparse representations, we successfully estimate poses to within  $15^\circ$  of the annotation in about 90% of the natural images. Without sparse representations, the success rate at this error tolerance is about 13% worse. To achieve the same success rate, we have to relax our error tolerance to almost  $22^\circ$ . Second, the performance of the system on the PIE images is not as good as on natural images. This likely indicates that the type of extreme lighting conditions present in the PIE images are not very common in our natural images.



Fig. 3. Examples of estimated poses.

## 6. CONCLUSION

We have presented here a method of computing sparse representations of facial images that preserve the information required to estimate pose with SVR. This both reduces the computation required to compute SVR and improves the accuracy of the results.

## 7. REFERENCES

[1] V. Blanz and T. Vetter. Face recognition based on fitting a 3d morphable model. *PAMI*, 25:1063–1074, 2003.

[2] R. Ishiyama and S. Sakamoto. Geodesic illumination basis: compensating for illumination variations in any pose for face recognition. In *Proc. 16th International Conference on Pattern Recognition*, volume 4, 2002.

[3] V. Kruger, S. Bruns, and G. Sommer. Efficient head pose estimation with gabor wavelet networks. In *British Machine Vision Conference*, 2000.

[4] S. Li, J. Yan, X.W. Hou, Z. Y. Li, and H. Zhang. Learning low dimensional invariant signature of 3-d object under varying view and illumination from 2-d appearances. In *ICCV*, 2001.

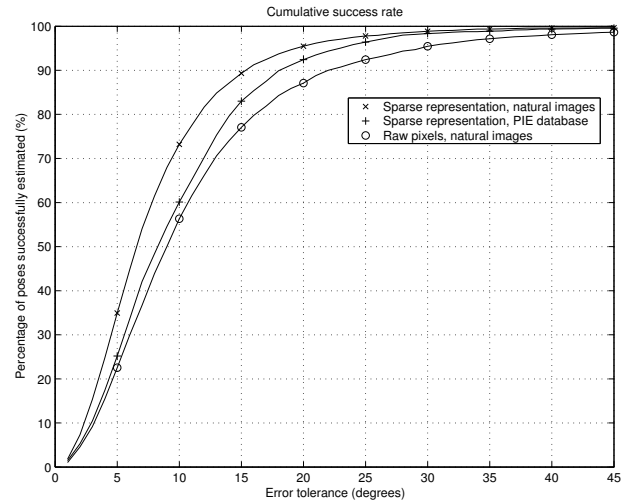


Fig. 4. Experimental results. This shows the frequency with which the difference between the estimated poses and the manual annotations fell within various error tolerances.

[5] Y. Li, S. Gong, and H. Liddell. Support vector regression and classification based multi-view face detection and recognition. In *Face and Gesture*, 2000.

[6] H. Moon, R. Chellappa, and A. Rosenfeld. 3d object tracking using shape-encoded particle propagation. In *ICCV*, 2001.

[7] H. Moon, R. Chellappa, and A. Rosenfeld. Optimal edge-based shape detection. *IEEE Transaction on Image Processing*, 11:1209–1226, 2002.

[8] P. J. Phillips, P. Grother, R. J. Micheals, D. M. Blackburn, E. Tabassi, and M. Bone. Face recognition vendor test 2002: Overview and summary. Technical report, 2003.

[9] T. Sim, S. Baker, and M. Bsat. The CMU pose, illumination, and expression (PIE) database of human faces. Technical Report CMU-RI-TR-01-02, The Robotics Institute, Carnegie Mellon University, January 2001.

[10] A.J. Smola and B. Schoelkopf. A tutorial on support vector regression. Technical Report NC2-TR-1998-030, NeuroCOLT2, 1998.

[11] V. Vapnik. *The Nature of Statistical Learning Theory*. Springer-Verlag, 1995.