

# Context-based additive logistic model for facial keypoint localization

Peng Li

p.li@cs.ucl.ac.uk

Jonathan Warrell

j.warrell@brookes.ac.uk

Jania Aghajanian

j.aghajanian@cs.ucl.ac.uk

Simon J. D. Prince

s.prince@cs.ucl.ac.uk

Department of Computer Science

University College London

Gower Street, London, United Kingdom

---

## Abstract

Facial keypoint localization is an important step for face recognition. The “Average of Synthetic Exact Filter (ASEF)” approach [1] finds a correlation filter for each training image and averages them together. The resulting classifier is efficient as the filtering can be implemented in the Fourier domain and performance is good for frontal images. However, it cannot cope with a range of poses. In this paper, we generalize this approach to find keypoints using a technique that (i) combines together information from training images in a more principled way than averaging, (ii) can be extended to form non-linear combinations of filters and (iii) can adapt based on context (e.g. pose). These innovations are presented within a greedy boosting-style probabilistic framework. We demonstrate state of the art performance of these algorithms using a challenging data set.

## 1 Introduction

Finding facial keypoints (corners of eyes, mouth, etc.) is important for face recognition, face tracking, expression recognition and other applications. For example, in recognition, keypoints are used to register the image to a common template [2] before inferences are made about identity. Recognition performance degrades if keypoints are not accurately localized [3]. Unfortunately, accurate localization of keypoints remains a difficult problem, due to wide variation in pose, facial expression, environmental conditions and the presence of occlusion (see [4] for a recent survey).

Previous methods for facial feature detection can be divided into two categories. *Generative methods* model the probability distribution of the observed data when the feature is present and absent, and compare the likelihood of the data under these two models using Bayes’ rule. *Discriminative models* directly model the binary posterior probability of the feature being present and parameterize this distribution as a function of the input data.

Examples of generative approaches include early methods that used heuristic rules and template matching [2, 5] as well as more recent Bayesian approaches [6, 7]. Discriminative approaches include neural networks [8], support vector machines [9, 10] and AdaBoost [11].

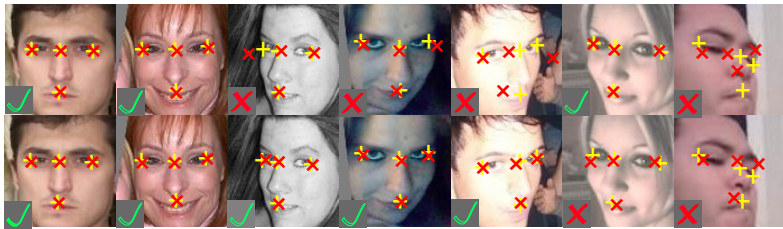


Figure 1: Top and bottom row show typical results for ASEF and our context-based additive logistic model (CALM) respectively. Yellow “+”s are ground truth keypoint positions and red “x”s are predicted keypoint positions. Our model is equal or better than ASEF on the first five images, slightly worse on the sixth. Both fail on the challenging final image.

In either case, it is typical to improve performance by exploiting the covariance of keypoint positions. For example, the position of the eyes tells us something about the position of the mouth. Wiskott et al. [16] proposed a distributed model of facial keypoints with a graph structure and these ideas have been revisited in the more modern form of pictorial structures ([8], [9], [13]). Although using the covariance improves performance, the final result still depends critically on the initial classifier.

Recently, Bolme et al. [9] proposed the “Average of Synthetic Exact Filters” (ASEF) for eye localization. A correlation filter is learned for each training image and these filters are averaged together. One advantage of this approach is that it uses the whole image in training rather than a subset of patches. Furthermore, ASEF is very fast in both training and testing because correlation can be computed efficiently in the Fourier domain. ASEF was shown to outperform Gabor Jets [16] and a Haar-based AdaBoost Cascade [13].

Despite these advantages, the ASEF algorithm is based on a single linear classifier. One challenge of detecting facial keypoints is the large variation in their appearance. For example, the eyes may be open or closed, the face pose and illumination may vary, the person may be smiling or angry. It is difficult for a single linear filter to capture this complexity. Moreover, it is not clear that the ASEF approach of averaging together individual correlation filters is the best way to form a single filter.

In this paper, we propose an additive logistic model to localize facial keypoints. In common with ASEF, the system exploits the entire image in training and can be implemented in the Fourier domain. However, unlike ASEF we train a set of filters using labeled training images rather than a single average filter and we combine them to optimize a well-defined cost function. Moreover, our classifier contains a bilinear component that causes the response to be modified by the pose (context). To further improve the keypoint localization we incorporate this classifier into a context dependent pictorial structure.

## 2 Average of Synthetic Exact Filters (ASEF)

We first briefly review the ASEF approach [9]. A filter  $\mathbf{H}$  is learnt from a set of training images  $\mathbf{I} = \{\mathbf{I}_n, n = 1, 2, \dots, N\}$  and their associated desired labels  $\mathbf{T} = \{\mathbf{T}_n | n = 1, 2, \dots, N\}$ . The desired outputs are synthetically generated images with a small two-dimensional Gaussian centred at the ground truth keypoint location (e.g. as in the top left of figure 2).

For a single image  $\mathbf{I}_n$  and a desired output  $\mathbf{T}_n$  the optimal filter  $\mathbf{H}_n$  can be computed

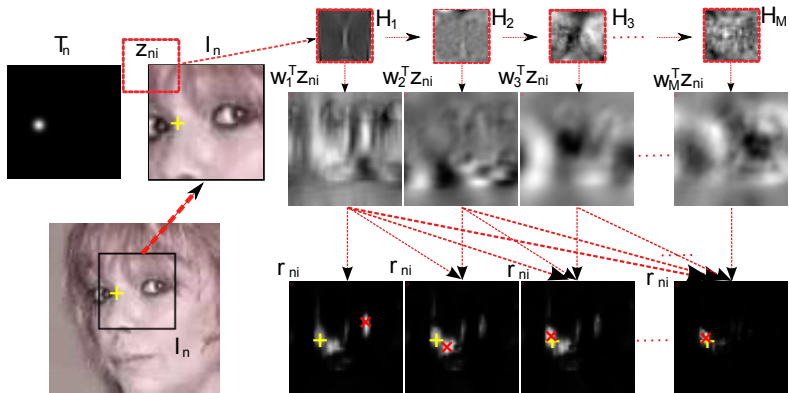


Figure 2: Training image  $I_n$  is a region from the original image (bottom left).  $T_n$  is the desired output for a keypoint (nose bridge) associated with  $I_n$ .  $I_n$  is correlated with a set of learned filters  $\mathbf{H}_1, \mathbf{H}_2, \dots, \mathbf{H}_M$  (top row). The concatenation of the filter elements forms  $\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_M$  respectively. When applied to the image, the filters give responses  $\mathbf{w}_1^T \mathbf{z}_{ni} \dots \mathbf{w}_M^T \mathbf{z}_{ni}$  respectively (middle row). The bottom row shows the final output of the ALM model  $r_n$  as we cumulatively add the filters. Yellow “+” cross denotes ground truth keypoint position. Red “x” is the predicted position (peak output position).

efficiently in the Fourier domain. Let  $\hat{\mathbf{I}}_n, \hat{\mathbf{T}}_n, \hat{\mathbf{H}}_n$  denote the 2D Fourier transforms of the image  $\mathbf{I}_n$ , desired correlation output  $\mathbf{T}_n$  and the filter  $\mathbf{H}_n$ . We now require that

$$\hat{\mathbf{T}}_n(\omega, \nu) = \hat{\mathbf{H}}_n^*(\omega, \nu) \hat{\mathbf{I}}_n(\omega, \nu), \quad (1)$$

where  $\omega$  and  $\nu$  index the spatial frequencies in the x and y directions and  $*$  denotes the complex conjugate. By dividing both sides by  $\hat{\mathbf{I}}_n(\omega, \nu)$  we get an expression for the complex conjugate of the Fourier transform  $\hat{\mathbf{H}}_n$  of the filter  $\mathbf{H}_n$ . This can be retrieved by taking the complex conjugate and performing an inverse Fourier transform. The average of synthetic exact filters is the average of the filters  $\mathbf{H}_n$  for each training image/desired output pair.

### 3 Additive Logistic Models (ALM)

The ASEF is simple to calculate and computationally efficient to apply. However, it is unclear that a single linear filter can cope with the pose, expression and illumination variation of real faces (figure 1) and even if it were, it is not obvious that an average of individually trained filters is the correct approach. To this end, we propose a classifier based on additive logistic models that (i) has a well defined cost function and (ii) can form non-linear combinations of several filters to capture variation of the target keypoint.

Let  $t_{ni}$  denote the label at the  $n$ 'th pixel of the  $i$ 'th image. The label  $t_{ni}$  equals one if the feature is certainly present and zero if certainly absent. Our goal is to find the posterior probability  $Pr(t_{ni} = 1 | z_{ni})$  of the label based on a data vector  $\mathbf{z}_{ni}$  extracted from the image at the current pixel. In this paper, the data vector will be the concatenated pixel values in a

square patch around the point. We define

$$Pr(t_{ni} = 1 | \mathbf{z}_{ni}) = r_{ni} = \sigma[a_{ni}] = \frac{1}{1 + \exp(-a_{ni})}, \quad (2)$$

where  $\sigma[\cdot]$  denotes the logistic sigmoid function which takes the activation  $a_{ni} \in [-\infty, \infty]$  and compresses it to the interval  $[0, 1]$ . The activation  $a_{ni}$  determines the tendency for the datum  $\mathbf{z}_{ni}$  to be considered as belonging to class one. It consists of an additive sum of a constant  $a_0$  and a series of functions  $f_m$  each of which acts on an associated linear projection of the datum  $\mathbf{w}_m^T \mathbf{z}_{ni}$  and has associated parameters  $\theta_m$ ,

$$a_{ni} = a_0 + \sum_{m=1}^M f_m[\mathbf{w}_m^T \mathbf{z}_{ni}, \theta_m]. \quad (3)$$

Here the functions  $f[\cdot]$  are arctan functions with parameters  $\theta = \{\alpha, \beta, \gamma\}$  so that

$$f(\mathbf{w}^T \mathbf{z} | \alpha, \beta, \gamma) = \alpha \arctan[\beta \mathbf{w}^T \mathbf{z} + \gamma]. \quad (4)$$

Since the data vector  $\mathbf{z}_{ni}$  is a square image patch, the projections  $\mathbf{w}_m$  can be interpreted as vectorized filters  $\mathbf{H}_m$  (figure 2). Each filter  $\mathbf{H}_m$  (or weight vector  $\mathbf{w}_m$ ) captures a characteristic of the facial keypoint and contributes to the overall activation  $a_{ni}$  which nonlinearly combines them. The target labels  $t_{ni}$  are very similar to the ‘desired outputs’ of the ASEF model. Hence the ALM model is similar in spirit to the ASEF approach, but has a well-defined cost function and can combine multiple filters to capture more complex characteristics of the facial keypoints.

### 3.1 Learning Additive Logistic Models

The additive logistic model can be learned by minimizing the negative log binomial posterior probability over patches  $\mathbf{z}_{ni}$  centered at  $N_K$  positions of  $N$  training images,

$$\arg \min_{a_0, \mathbf{w}_1 \dots \mathbf{w}_M, \theta_1 \dots \theta_M} L = \arg \min_{a_0, \mathbf{w}_1 \dots \mathbf{w}_M, \theta_1 \dots \theta_M} \sum_{n=1}^N \sum_{i=1}^{N_K} -\log[Pr(t_{ni} | \mathbf{z}_{ni})], \quad (5)$$

where  $t_{ni}$  is the desired output of the square patch  $\mathbf{z}_{ni}$  centered at the  $i$ ’th location in the  $n$ ’th training face image  $I_n$ .

We take a boosting-style sequential approach to optimize this cost function in which we add the functions  $f_1 \dots f_M$  one at a time. In the first stage we optimize over parameters  $a_0, \mathbf{w}_1, \theta_1$  to fit a model with activation:  $a_{ni} = a_0 + f_1(\mathbf{w}_1^T \mathbf{x}_{ni}, \theta_1)$ . At the second stage, we fix  $\mathbf{w}_1$  and  $\theta_1$  and optimize over parameters  $a_0, \mathbf{w}_2$  and  $\theta_2$  to fit a model with activation:  $a_{ni} = a_0 + f_1(\mathbf{w}_1^T \mathbf{x}_{ni}, \theta_1) + f_2(\mathbf{w}_2^T \mathbf{x}_{ni}, \theta_2)$ . At the  $m$ ’th stage, we fix  $\mathbf{w}_1 \dots \mathbf{w}_{m-1}$  and  $\theta_1 \dots \theta_{m-1}$  and optimize over parameters  $a_0, \mathbf{w}_m$  and  $\theta_m$  and so on (see figure 2).

At each stage we optimize the binomial log likelihood criterion,  $L$  using a gradient descent method which requires the first derivatives of  $L$  with respect to parameters  $\phi_m = \{a_0, \mathbf{w}_m, \theta_m\}$  of  $\mathbf{a}_{ni}$ . These are calculated using the relations

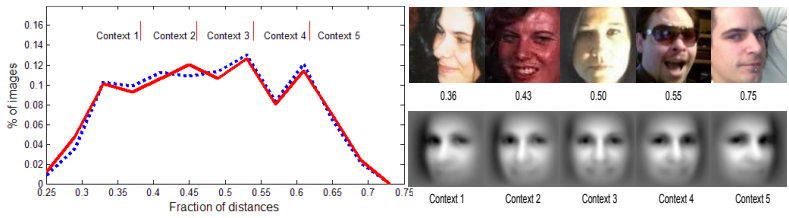


Figure 3: Distribution of training (red) and test images (blue) over different contexts (pose categories). Pose is quantified as the ratio of the distance between left eye and nose bridge to interocular distance. Top right are the samples from each context. Bottom right are the mean faces of each context.

$$\begin{aligned}
 \frac{\partial L}{\partial \mathbf{w}_m} &= \sum_{n=1}^N \sum_{i=1}^{N_K} \frac{\alpha_m \beta_m (r_{ni} - t_{ni}) \mathbf{z}_{ni}}{1 + (\beta_m Q_{ni} + \gamma_m)^2} & \frac{\partial L}{\partial \alpha_m} &= \sum_{n=1}^N \sum_{i=1}^{N_K} (r_{ni} - t_{ni}) \arctan(\beta_m Q_{ni} + \gamma_m) \\
 \frac{\partial L}{\partial \beta_m} &= \sum_{n=1}^N \sum_{i=1}^{N_K} \frac{\alpha_m Q_{ni} (r_{ni} - t_{ni})}{1 + (\beta_m Q_{ni} + \gamma_m)^2} & \frac{\partial L}{\partial \gamma_m} &= \sum_{n=1}^N \sum_{i=1}^{N_K} \frac{\alpha_m (r_{ni} - t_{ni})}{1 + (\beta_m Q_{ni} + \gamma_m)^2} \\
 \frac{\partial L}{\partial a_0} &= \sum_{n=1}^N \sum_{i=1}^{N_K} (r_{ni} - t_{ni}) & & (6)
 \end{aligned}$$

where  $Q_{ni} = \mathbf{w}_m^T \mathbf{z}_{ni}$  is the response of filter  $\mathbf{H}_m$  at the  $i$ 'th location of image  $I_n$  which can be calculated efficiently in the Fourier domain. In fact, the gradient with respect to  $\mathbf{w}_m$  can also be computed using convolution, but details of this are omitted to save space.

## 4 Context-based Additive Logistic Models (CALM)

The appearances of the facial keypoints vary considerably (figure 1) due to factors like expression, lighting and pose which we term *contexts*. If we have information about the context, we can adapt the model accordingly. To this end, we propose a context-based Additive Logistic Model (CALM) which modifies its responses based on the context. In this paper, we illustrate the properties of the proposed algorithm by taking face pose as the context, which was estimated using the approach of [10].

For each training image  $I_n$ , the context  $c_n \in \{1 \dots J\}$  is discrete. Equation 3 now depends on the context  $j$  so that

$$a_{ni} = \sum_{j=1}^J Pr(c_n = j) [a_{0j} + \sum_{m=1}^M f_{mj}(\mathbf{w}_{mj}^T \mathbf{z}_{ni}, \theta_{mj})] \quad (7)$$

where we incorporate uncertainty in the context estimate using the posterior  $Pr(c_n = j)$  returned by the pose classifier. Note that the context not only determines the functions parameters  $\theta_{mj}$ , but also the projections  $\mathbf{w}_{mj}^T \mathbf{z}_{ni}$  of the data: we measure the same aspects of the image, but we interpret these measurements differently depending on the situation.

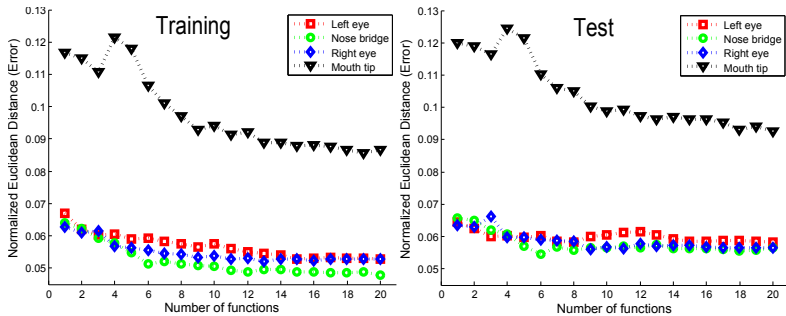


Figure 4: Average detection error of ALM for all four keypoints on training and test data set.

The context-based additive logistic model is again learned stagewise by minimizing the negative log binomial posterior probability:

$$\arg \min_{a_{01}, \dots, a_{0C}, \mathbf{w}_{11}, \dots, \mathbf{w}_{1C}} L = \arg \min_{a_{01}, \dots, a_{0C}, \mathbf{w}_{11}, \dots, \mathbf{w}_{1C}} \sum_{n=1}^N \sum_{i=1}^{N_K} -\log[Pr(t_{ni} | \mathbf{z}_{ni})], \quad (8)$$

where we now add a subset of classifiers at each stage. For the  $m$ 'th round we optimize over parameters  $\mathbf{w}_{m1}, \dots, \mathbf{w}_{mJ}, a_{01}, \dots, a_{0J}, \theta_{m1}, \dots, \theta_{mJ}$ .

## 5 Context-based pictorial structure model

Although we model the appearance of facial keypoints independently using a discriminative model, it has been shown that exploring the relations of the keypoint positions in a generative model is able to improve keypoint localization [8], [9]. The single tree-based pictorial structure [8] models the joint positions pairs of facial keypoints as a Gaussian. Everingham et al. [9] proposed a mixture of Gaussian trees to model the joint position of the facial keypoints, with mixture components corresponding to frontal faces and slight profile faces. Pose information is not used explicitly in this model and their AdaBoost-based keypoint detectors are trained using only frontal or near frontal faces.

In our case, we have access to robust pose information, and we exploit this by building a context-based pictorial structure model: we learn a separate pictorial structure model for each of the pose clusters. In the localization stage, we estimate the MAP context of the test image and use the corresponding tree to help infer the keypoint positions.

## 6 Results

We evaluate our proposed method on a subset of the UCL database [10] which consists of  $100 \times 100$  pixel images captured from on-line dating websites and contains significant variation in pose, lighting and expression. In each case, the faces were automatically detected by a commercial face detector. The face image was normalized relative to the scale and orientation of the detected region such that the hair region was included.

We quantized the pose using the ratio of the distance between the left outer eye corner and the nose bridge and the distance between the two outer eye corners. This fraction changes

from zero to one as the face moves from left-profile to right profile. The database was divided into 5 pose clusters according to this fraction (figure 3). We randomly selected 700 images from each category and use 500 of them for training and the other 200 for test.

Each image was normalized by passing it through a  $\log$  function ( $\log(x+1)$ ) to suppress the effect of shadows and lighting, then a cosine window is applied to the image following [2] to reduce edge artifacts. Cosine windows were also applied to each separate keypoint region and these regions were finally normalized to zero mean and unit standard deviation.

Four keypoints (two outer eye corners, nose bridge and mouth tip) were manually labeled to provide ground truth. The search region for each keypoint was set to within 22 pixels of the mean keypoint coordinate. The filter size and the standard deviation of the Gaussian shaped function defining the desired output  $t_n$  were optimized by cross validation.

We compared ALM with three other algorithms: ASEF [2], AdaBoost [5] and the Bayesian approach of [6]. The patch sizes used by AdaBoost with 1000 Haar-feature-based weak classifiers (not a cascade) and the Bayesian approach are kept the same as that of ALM. The ASEF filter was trained using the full  $45 \times 45$  image region as shown by the image region  $I_n$  of figure 2. Evaluation of the eye detection algorithms is based on the distance from the predicted position to the manually labeled position, normalized by the inter-ocular distance (two eye corners here) following [2, 6].

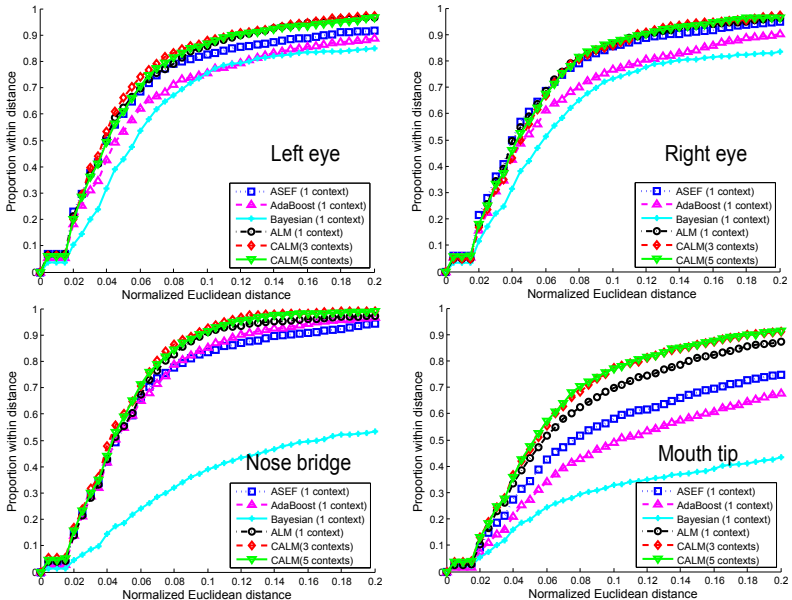


Figure 5: Cumulative errors of ASEF [2], AdaBoost [5], Bayesian [6] and ALM assuming different number of contexts in terms of normalized Euclidean distance.

## 6.1 Additive Logistic Model (ALM)

Figure 4 illustrates the average positional error of four facial keypoints on the training and test data sets. In both cases the detection error decreases gradually as more filters are included. The decrease is not consistent in training because our target is to minimize the

Keypoints	1. (Left Eye)	2. (Right Eye)	3. (Nose Bridge)	4. (Mouth Tip)
ASEF [2]	0.069(0.003)	0.062(0.002)	0.065(0.002)	0.128(0.004)
AdaBoost [5]	0.086(0.003)	0.086(0.003)	0.065(0.002)	0.160(0.005)
Bayesian [5]	0.106(0.004)	0.117(0.005)	0.268(0.007)	0.265(0.006)
ALM(1 context)	0.056(0.002)	<b>0.058(0.002)</b>	0.057(0.002)	0.092(0.003)
ALM(3 contexts)	<b>0.054(0.002)</b>	0.059(0.002)	<b>0.050(0.001)</b>	<b>0.081(0.003)</b>
ALM(5 contexts)	0.057(0.002)	0.059(0.002)	0.051(0.001)	<b>0.081(0.003)</b>

Table 1: Result of localization error (mean and standard error) in terms of normalized Euclidean distance on UCL data subset.

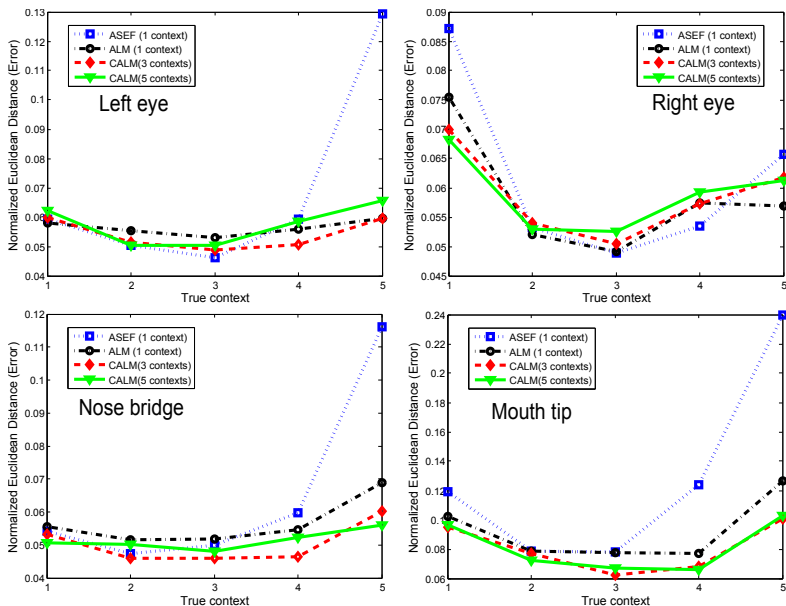


Figure 6: Performance of ASEF and ALM assuming different number of contexts on the test images from different contexts (poses).

negative log binomial posterior probability rather than directly minimize the positional error.

Table 1 shows average errors over 1000 test images by ASEF [2], AdaBoost [5], the Bayesian approach [5] and ALM (1–context). Cumulative error distributions are plotted in figure 5. Our model outperforms the other algorithms for all four keypoints, and especially for the nose bridge and mouth tip. This may be because eye corners exhibit less variation compared to the nose bridge and mouth tip and are easier to characterize. In particular, linear models (ASEF, Bayesian) may find it very difficult to classify these features under wide pose variations. Our model which is based on a non-linear combination of many filters is more expressive. This may also account for the fact that AdaBoost (non-linear) outperforms the Bayesian approach (linear) in this dataset (in contradiction to [5]).

As in the original experiments of Bolme et al [2] ASEF outperforms both AdaBoost and the Bayesian approach. This may be because it is based on convolution and hence uses *all*



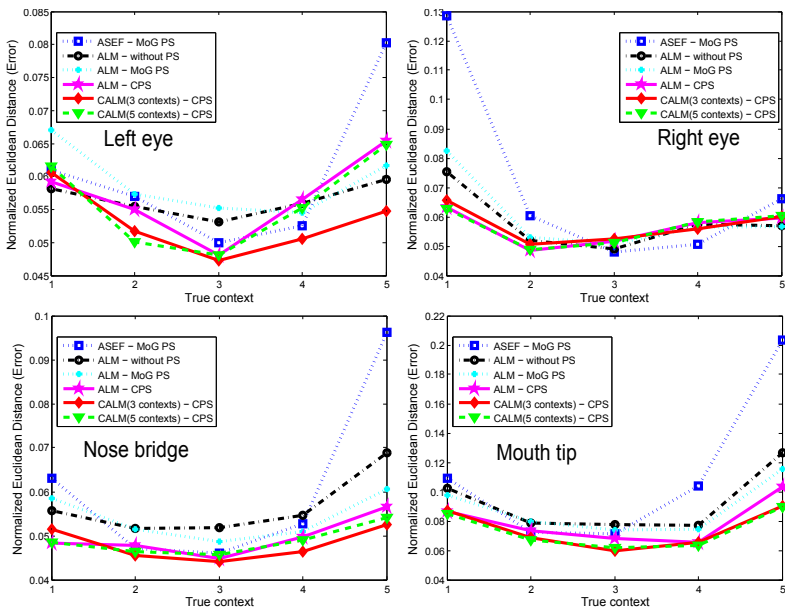


Figure 7: Average localization error of different algorithms with pictorial structure on the test images from different contexts (poses).

of the patches in the training images rather than just a subset. Our algorithm, is non-linear and also exploits all of the patches and combines the best of both approaches.

## 6.2 Context Based Additive Logistic Models (CALM)

Figure 5 and Table 1 also show results for our approach with 3 and 5 contexts. Introducing context (pose-dependent models) generally decreases the error and particularly helps for the more difficult features (mouth and nose bridge). Figure 6 plots the positional error against the true context: for frontal and near-frontal faces (contexts 2-4) both ASEF and ALM works well. In fact, ASEF is a little bit better than ALM in detecting the two eye corners in the frontal faces (context 3). However, performance of ASEF degrades significantly when the pose change is large (context 1 and 5). In contrast, our proposed CALM model still performs reasonably well by exploring the characteristics of each keypoint on different contexts (poses here). This supports our claim that context-based CALM is better than ALM model assuming only 1 context.

## 6.3 Context-based pictorial structure

In Figure 7 we compare the localization error for the context-based pictorial structure (CPS) with a mixture of Gaussian tree-based pictorial structure (MoG PS) [17]. We also implemented a pictorial structure based on a single tree [8] (not shown) but found that it degraded performance: it cannot capture the relative positions of the features with large pose changes.

The data shows that when we use the MoG PS the error decreases in the two difficult keypoints (nose bridge and mouth tip) but is a little worse in the two eye corners. When

the context-based pictorial structure is applied to ALM model, we observed improvement over all four keypoints: context information helps localize keypoints by choosing the most appropriate pictorial structure. Best performance is achieved by combining the context-based ALM model and the context-based pictorial structure. We also compare the ALM model and ASEF models incorporated in an MoG PS. Our method still outperforms the ASEF approach under these circumstances.

## 7 Conclusion

We have proposed an additive logistic model for localizing facial keypoints. It should also be noted that this model has close connections with other learning approaches: it resembles boosting approaches in that it incrementally adds classifiers, and for the arc-tangent functions used here the final classifier resembles a multi-layer perceptron in which hidden units are sequentially added and optimized.

Our model benefits from (i) using all of the patches in the training data and (ii) combining many linear responses in a non-linear function to better characterize the overall appearance variation of facial keypoints. We also demonstrate that performance can be further improved if we have an external source of context information (here pose).

## References

- [1] J. Aghajanian and S. J. D. Prince. Face pose estimation in uncontrolled environments. In *BMVC*, 2009.
- [2] D. S. Bolme, B. A. Draper, and J. R. Beveridge. Average of synthetic exact filters. In *CVPR*, pages 2105–2112, 2009.
- [3] P. Campadelli, R. Lanzarotti, and G. Lipori. Precise eye localization through a general-to-specific model definition. In *BMVC*, page I:187, 2006.
- [4] P. Campadelli, R. Lanzarotti, and G. Lipori. *The Fundamentals of Verbal and Non-verbal Communication and the Biometrical Issue*, volume 18 of *NATO Science Series*, chapter Eye localization: a survey, pages 234–245. IOS Press, Amsterdam, May 2007.
- [5] D. Cristinacce and T. Cootes. Facial feature detection using AdaBoost with shape constraints. In *BMVC*, pages 231–240, 2003.
- [6] M. Everingham and A. Zisserman. Regression and classification approaches to eye localization in face images. In *FGR*, pages 441–448, 2006.
- [7] M. Everingham, J. Sivic, and A. Zisserman. “Hello! My name is... Buffy” – automatic naming of characters in TV video. In *BMVC*, 2006.
- [8] P. F. Felzenszwalb and D. P. Huttenlocher. Pictorial structures for object recognition. *IJCV*, 61(1):55–79, 2005.
- [9] P. Li and S. J. D. Prince. Joint and implicit registration for face recognition. In *CVPR*, pages 1510–1517, 2009.

- [10] A. M. Martínez. Recognizing imprecisely localized, partially occluded, and expression variant faces from a single sample per class. *TPAMI*, 24(6):748–763, 2002.
- [11] M. J. T. Reinders, R. W. C. Koch, and J. J. Gerbrands. Locating facial features in image sequences using neural networks. In *AFGR*, pages 230–235, 1996.
- [12] J. Rurainsky and P. Eisert. Eye center localization using adaptive templates. In *CVPRW'04*, volume 5, page 67, 2004.
- [13] X. Y. Tan, F. Song, Z. H. Zhou, and S. C. Chen. Enhanced pictorial structures for precise eye localization under uncontrolled conditions. In *CVPR*, pages 1621–1628, 2009.
- [14] M. Turk and A. Pentland. Face recognition using eigenfaces. In *CVPR*, pages 586–591, 1991.
- [15] P. Viola and M. J. Jones. Robust real-time face detection. *IJCV*, 57(2):137–154, May 2004.
- [16] L. Wiskott, J. M. Fellous, N. Kruger, and C. von der Malsburg. Face recognition by elastic bunch graph matching. *TPAMI*, 19(7):775–779, 1997.
- [17] A. L. Yuille, D. S. Cohen, and P. W. Hallinan. Feature extraction from faces using deformable templates. *IJCV*, 8(2):99–111, August 1992.