A Bayesian Occlusion Model for Sequential Object Matching

Toni Tamminen and Jouko Lampinen
Laboratory of Computational Engineering
Helsinki University of Technology
P.O. Box 9203, 02015 HUT, Finland
{toni.tamminen, jouko.lampinen}@hut.fi

Abstract

We consider the problem of locating instances of a known object in a novel scene by matching the fiducial features of the object. Our approach to the problem consists of two parts: a model for the appearance of the features and a model for the shape of the object. We then bind these parts together in a Bayesian framework and match the features sequentially, using the information about the locations of previously matched features. Into this matching system we add a Bayesian model for dealing with features that are not detected due to occlusion or abnormal appearance. Our system yields promising results, losing little matching accuracy even for heavily occluded objects.

1 Introduction

Occlusion is one of the major challenges in computer vision. Especially in feature based recognition it is very problematic - if the interesting features of an object are not detected, it is very hard to locate or recognize the object. For rigid object models, proposed approaches include employing edge detection and the Haussdorf distance [11], directed edges and other similarity measures [12], and intensity-based matching in a Bayesian framework [14]. These approaches have obtained very good results, but for deformable models the task is more challenging. Solutions have been proposed for dynamic tracking problems [16] as well as static situations [5], but also these approaches usually deal with occlusions with a "rigidness parameter" controlling the extent of allowed deformations instead of applying a formal occlusion model.

In this paper, we present a novel Bayesian occlusion model incorporated into a matching system capable of dealing with complex features and object shapes. A key feature of our matching model is its sequential nature, in which the information about the location of the previously matched features aids in matching of the following ones. Our results indicate that the system is capable of locating even heavily occluded objects with little decrease in matching performance.

This paper is organized as follows. Sections 2 and 3 describe our feature and object models. Section 4 combines these models and our occlusion model in a Bayesian way to produce a joint representation of objects. Section 5 illustrates the sequential matching

scheme, while Section 6 presents the matching results obtained by the system. Section 7 concludes the work.

2 Feature Model

All natural images contain clutter and noise. To make the features of interest more distinctive in an image, we first transfer the observed image $\mathbf{I}$ into feature space $\mathbf{T}$, $\mathbf{I} \mapsto \mathbf{T}$, so that each image pixel $\mathbf{I}(u,v)$ has associated features $\mathbf{T}(u,v)$. As the transformation $\mathbf{I} \mapsto \mathbf{T}$ we employ a Gabor filter bank with 3 frequencies and 6 orientations. Gabor filters are direction-sensitive edge detectors well-suited to feature matching tasks [3]. The filter responses are stacked as vectors, or jets, to produce $\mathbf{T}(u,v)$. This approach is similar to the Local Jet framework of Koenderink and Doorn [9].

To find the locations of the features of an object in an image we need to compare the perceived jets $\mathbf{T}(u,v)$ and the jets we would expect the features to have. We estimate the target jets by assuming that the distribution of the amplitude and phase jets in the feature space both follow a Gaussian distribution and determine the distribution parameters $\mathbf{G} = \{g_1, ..., g_m\}$ by measuring the jets at the feature locations from a set of manually pre-annotated faces. To improve contrast-independence, we normalize the jets, but since total contrast-independence causes the system to be sensitive to faint patterns and noise in uniform areas, we add a Gaussian term measuring the energy of the Gabor jet. Finally, since the amplitude and phase distributions are high-dimensional ($d = 17$), and annotated training images are usually limited in number due to the effort required to produce them, we regularize the model by adding a constant ridge term ε_G to the diagonals of the covariance matrices. That is, the covariance matrices become $\Sigma^* = \Sigma + \varepsilon_G \mathbf{I}$. This ridge parameter controls the steepness of the similarity function.

By combining the amplitude, phase, and energy components, we get the total similarity between the perceived jet $\mathbf{T}(u,v)$ and the distribution of the jet corresponding to the ith feature:

$$S(\mathbf{T}(u,v), g_i) \propto \mathrm{N}^*(G_{\text{amp}} | \mu_{\text{amp}}, \Sigma_{\text{amp}}) \cdot \mathrm{N}^*(G_{\text{phase}} | \mu_{\text{phase}}, \Sigma_{\text{phase}}) \cdot \mathrm{N}^*(G_{\text{energy}} | \mu_{\text{energy}}, \sigma^2_{\text{energy}}), \tag{1}$$

where N^* is the unnormalized Gaussian density function, $G_{\text{amp}}, G_{\text{phase}}$ and G_{energy} are the Gabor jet properties corresponding to $\mathbf{T}(u,v)$, and $\mu_{\text{amp}}, \mu_{\text{phase}}, \mu_{\text{energy}}, \Sigma_{\text{amp}}, \Sigma_{\text{phase}}$ and σ_{energy} the means and (co)variances of the corresponding distribution g_i. The similarity measure is illustrated in Figure 1. It can be seen that he similarity fields are multimodal, and thus other information besides the feature model is required for successful matching.

The similarity measure presented here closely resembles the one by Wiskott *et al.* [15] both in content and in performance. Both of them are rather *ad hoc*, but this is the case for all such feature similarity measures. As long as there are no generative models for natural images, engineering solutions such as these will have to do.

3 Object Model

The object model is learned from the set of training shapes $\mathbf{Y} = \{Y_1, ..., Y_m\}$, which are the annotations used in the learning of the features. To eliminate pose effects and the random asymmetry of human faces, we use a mirrored replicate of each training shape as part of

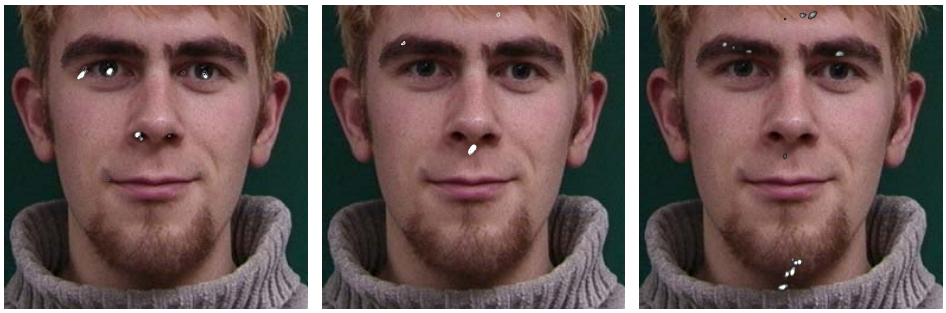

Figure 1: Sample feature similarity fields. The target features are the outer corner of the leftmost eye (left image), the point between the nostrils (center image) and the tip of the chin (right image). Note how the fields are multimodal, with peaks in several distinct locations.

the training data set. An example of a training shape distribution is illustrated in Figure 2.

We consider an object shape to be the sum of a basic object and variations from this shape. A simple representation of this is a Gaussian distribution with the basic shape represented by the mean and the variations by the covariance. Since we do not have any real prior information about the covariance of the features, we set a vague congujate inverse-Wishart prior on it:

$$\Sigma \sim \text{Inv} - \text{Wishart}_{\nu_0}(\Lambda_0^{-1}),\ \mu|\Sigma \sim \text{N}(\bar{\mathbf{Y}}, \Sigma/\kappa_0), \tag{2}$$

where ν_0 and Λ_0 describe the degrees of freedom and the scale matrix for the inverse-Wishart distribution, and κ_0 is the number of prior measurements. The posterior distribution of the distribution parameters is from the normal-inverse-Wishart family with parameters [7]

$$\mu_m = \bar{\mathbf{Y}},\ \kappa_m = \kappa_0 + m,\ \nu_m = \nu_0 + m,\ \Lambda_m = \Lambda_0 + \sum_{i=1}^{m} (\mathbf{Y}_i - \bar{\mathbf{Y}})(\mathbf{Y}_i - \bar{\mathbf{Y}})^T. \tag{3}$$

The quantity of our interest, the predictive distribution of a new shape $\mathbf{x}$ given the training shapes, can be computed with the result

$$p(\mathbf{x}|\mathbf{Y}) = \int p(\mathbf{x}|\xi)p(\xi|\mathbf{Y})d\xi = t_{\nu_m - d + 1}(\mu_m, \Lambda_m(\kappa_m + 1)/(\kappa_m(\nu_m - d + 1))), \tag{4}$$

where d is the dimension of the distribution ($d = 116$ in our case for 58 features) and ξ denotes collectively the parameters μ and Σ. For our sequential sampling scheme we need to be able to compute the conditional distributions of this distribution, as described in Section 5. The parameters of the prior distributions were set such that we could approximate this t-distribution with a Gaussian one with the number of training samples we had ($m = 37,\ \kappa_0 = 1,\ \nu_0 \approx 100$). The conditional distributions can easily be computed from the Gaussian approximation, and thus our final model for the shape of the object to be matched is

$$p(\mathbf{x}|\mathbf{Y}) \approx \text{N}(\mu_m, \Lambda_m(\kappa_m + 1)/(\kappa_m(\nu_m - d + 1))). \tag{5}$$

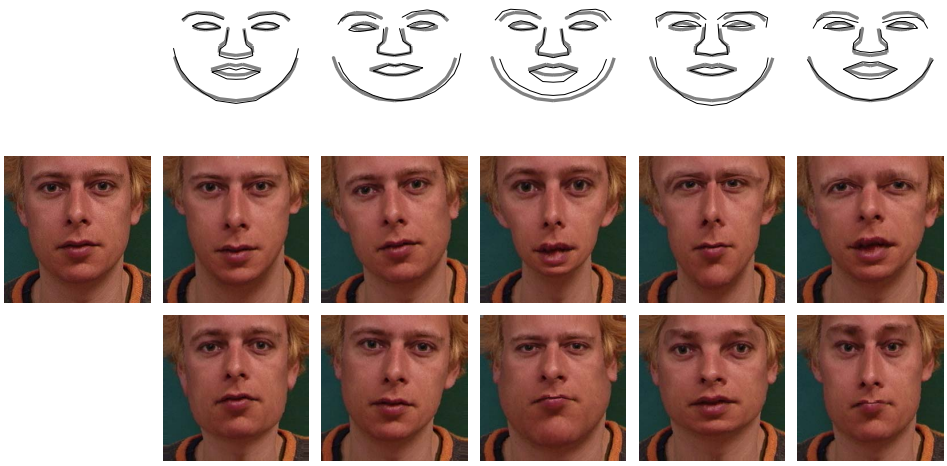

Figure 2: A sample training shape distribution. In the upper row, the thick gray graphs show the mean shape and the thinner black graphs the leading eigenvectors added to the mean. In the other rows, the face on the left has been morphed according to the principal components, both in the positive (middle row) and negative (lower row) directions. Components 1 and 2 appear to be related to rotations, while components 3, 4, and 5 are shape-related.

4 Probability Model

The previous two sections described our feature and object shape models, which we combine with a Bayesian probability model. A similar Bayesian approach to image analysis has been proposed by Li *et al.* [10], although their aim is the classification of images according to the objects contained in them rather than the precise matching of the objects.

In the Bayesian point of view all observed and unobserved quantities are considered random variables following some distributions [7]. Our observed variables are the feature image $\mathbf{T}$ and the training features $\mathbf{G}$ and object shapes $\mathbf{Y}$. Our unobserved variables are the locations of the n target features arranged into a planar configuration, $\mathbf{x} = \{x_1, \ldots, x_n\}$ as well as the shape model hyperparameters ξ.

Without the occlusion model our aim is to estimate the posterior distribution of the feature locations given the image and the training data, with the hyperparameters integrated out:

$$p(\mathbf{x}|\mathbf{T},\mathbf{G},\mathbf{Y}) \propto p(\mathbf{T}|\mathbf{x},\mathbf{G}) \int p(\mathbf{x}|\xi)p(\xi|\mathbf{Y})d\xi, \tag{6}$$

where $p(\mathbf{T}|\mathbf{x},\mathbf{G})$ is the image likelihood and $\int p(\mathbf{x}|\xi)p(\xi|\mathbf{Y})d\xi$ the prior, composed of the object shape and hyperprior parts. The likelihood measures the probability of observing the feature image $\mathbf{T}$ given the feature locations $\mathbf{x}$ and the training features $\mathbf{G}$. The object shape prior is the predictive distribution of a new object shape given the training shapes. That is, we consider $\mathbf{x}$, the feature configuration to be estimated, a new sample from the distribution of object shapes estimated from the training data. Here we have made some independence assumptions, namely that the training features $\mathbf{G}$ only affect

the image likelihood and the training shapes $\mathbf{Y}$ only affect the prior.

Our object model can be used directly as the prior part, but the feature similarity measure can not directly be interpreted as the likelihood $p(\mathbf{T}|\mathbf{x},\mathbf{G})$. As in practice it is extremely difficult to assess the probability of observing an image given a feature configuration, we make the simplifying assumption that the likelihoods of the transformed pixels of the image are independent of each other and dependent only on the individual feature locations x_i and training features g_i pertaining to the ith feature, and approximate the individual likelihoods with their feature similarities so that $p(\mathbf{T}_i|x_i,g_i) \approx S(\mathbf{T}(x_i),g_i)$. The likelihood of observing an image given a feature configuration $\mathbf{x}$ is given by multiplying the feature similarities at the feature locations specified by the configuration. The similarities are computed for all pixels of the image so that the features can in principle be located anywhere in the image. With this likelihood, we can compute posterior distributions of feature configurations with (6). In practice, due to the sequential nature of the actual matching, we are interested in the conditional posteriors

$$p(x_i|\mathbf{T},\mathbf{x}',\mathbf{G},\mathbf{Y}) \propto p(\mathbf{T}_i|x_i,g_i)\int p(x_i|\mathbf{x}',\xi)p(\xi|\mathbf{Y})d\xi, \tag{7}$$

where $\mathbf{x}' = x_{1,\ldots,i-1}$ is the set of features matched before the ith one.

To include the possibility of occlusion into the matching model, we add a vector of indicator variables γ such that

$$\gamma_i = 1, \text{ if the } i\text{th feature is detected} \tag{8}$$
$$\gamma_i = 0, \text{ if the } i\text{th feature is not detected} \tag{9}$$

Now our aim is to infer the marginal posterior distribution of the ith feature:

$$p(x_i|\mathbf{T},\mathbf{x}',\mathbf{G},\mathbf{Y},\gamma') = \int p(x_i,\gamma_i|\mathbf{T},\mathbf{x}',\mathbf{G},\mathbf{Y},\gamma')d\gamma_i, \tag{10}$$

where $\gamma' = \gamma_{1,\ldots,i-1}$ denotes whether the features matched before the ith one were detected. Since there are only two possible values for γ_i, the integral can be written as the sum

$$p(x_i|\mathbf{T},\mathbf{x}',\mathbf{G},\mathbf{Y},\gamma') = p(x_i,N_i|\mathbf{T},\mathbf{x}',\mathbf{G},\mathbf{Y},\gamma') + p(x_i,\bar{N}_i|\mathbf{T},\mathbf{x}',\mathbf{G},\mathbf{Y},\gamma'), \tag{11}$$

where we have denoted $\gamma_i = 1$ with N_i and $\gamma_i = 0$ with $\bar{N}_i$. With Bayes' theorem, the posterior (10) can be written as

$$p(x_i|\mathbf{T},\mathbf{x}',\mathbf{G},\mathbf{Y},\gamma') \propto [p(\mathbf{T}_i|x_i,g_i,N_i)P(N_i|\gamma') + p(\mathbf{T}_i|x_i,g_i,\bar{N}_i)P(\bar{N}_i|\gamma')]\int p(x_i|\mathbf{x}',\xi)p(\xi|\mathbf{Y})d\xi, \tag{12}$$

where we have assumed that the object prior is independent of feature detection and that the prior probabilities of detection and no-detection are dependent only on the detections of the previously matched features, whereas the likelihood is assumed not to depend on the previous detections. Furthermore, since we do not have a model for the interdependence of the occlusions, we assume that the detection probabilities are *a priori* independent of the previous detections, $P(N_i|\gamma') = P(N_i)$. For known occlusion configurations (for example, if we knew that one horizontal half of the object was occluded) such a model could easily be included.

The difficult part of the model is $p(\mathbf{T}_i|x_i,g_i,\bar{N}_i)$, the likelihood of observing the image when the feature is not detected. We use a flat likelihood - since the feature is not detected,

we get no information about its location from the image. The relative level of the flat likelihood is another problem, as it determines the balance of possible detections and no-detections in the image. We used the mean of the likelihood of the feature over the whole image, which seemed to work rather well.

5 Sampling

Our posterior distributions can not be evaluated in closed form as the likelihood can only be computed numerically, but we can obtain samples from the distribution and estimate the quantities of interest such as the posterior mean from the samples. Without the occlusion model we have used Markov chain Monte Carlo (MCMC) methods [8] to produce the samples. Introducing the possibility of occlusion increases the multimodality of the posterior (the posterior of object location can vary lot according to which features are assumed to be detected), which lowers the efficiency of these methods. In theory we could sample also over the detection variables γ with MCMC, but this would greatly increase the dimensionality and the computational requirements of the problem. Instead, we use sequential Monte Carlo (SMC) [4] to draw the samples, especially as according to our experience SMC seems to explore multimodal posteriors better than MCMC methods (the Gibbs and Metropolis-Hastings samplers change mode very seldom if the modes are distinct). SMC methods have usually been used for dynamic problems such as tracking, but it is also possible to sample from static posteriors with them [1].

Bayesian SMC algorithms represent the posterior as a weighted set of particles (θ_k, w_k). The particles are drawn from a proposal distribution $\pi(\theta_{k+1}|\theta_k)$, after which the weights are computed with the ratio of the target distribution and the proposal distribution at the sampled point:

$$w_{k+1} = \frac{p(y|\theta_{k+1})p(\theta_{k+1}|\theta_k)}{\pi(\theta_{k+1}|\theta_k)}. \tag{13}$$

Usually either the prior $p(\theta_{k+1}|\theta_k)$ or the likelihood $p(y|\theta_{k+1})$ is used as the proposal, which simplifies the weight equation. After all particles have been updated, a resampling step according to the particle weights is often performed - since some particles tend to have almost zero weights and thus do not affect any estimates, it is sensible to reallocate these to the particles with weights larger than zero.

Our target posterior distribution is given in (12). If a feature is visible in the image, the corresponding likelihood is peaked, and a reasonable proposal is the likelihood itself (which is also used in the system without the occlusion model). If a feature is not visible, the likelihood is flat, and we would like to use the prior as the proposal. One solution is to use a mixture proposal distribution composed of both the prior and likelihood terms $\pi(\theta_{k+1}|\theta_k) = \phi\, p(y|\theta_{k+1}) + (1-\phi)p(\theta_{k+1}|\theta_k)$, where ϕ is the mixing ratio between the two proposals [6]. To make both terms equally meaningful, both the likelihood and the prior must be normalized. To reduce computational load, we only evaluate the likelihood and prior within some "reasonable" distance from the prior mean, for example 3 σ's. Scaling of the object is handled by estimating the scale from the previously matched features and modifying the prior accordingly.

Since we do not know which features are visible and which are not, we randomize the matching order of the features. This is done by randomly assigning each particle a feature from the list of unmatched features after the resampling step. As the occluded features

have lower likelihood values than the visible ones and thus lower posterior probabilities, they receive lower weights, which causes the visible features to be (mostly) matched first. For the first feature there are no previously matched features, and thus we draw it directly from the likelihood, implying a flat prior.

6 Results

We evaluated the performance of the matching system by testing it with both simulated and real occlusions. For the simulated occlusions, we used the IMM-DTU database [13] consisting of 37 annotated facial images. An occlusion was first generated for each image by setting the pixel values of the right half of the image to the mean pixel value of the image - over half of the features were thus occluded (the ones near the edge can also be considered occluded, as the very strong edge confuses the Gabor filters). Each image was then matched in turn by using the other 36 images as the training data. The Λ_0 hyperparameter was set such that the standard deviation for the final features was around 4 pixels. Sample results from matching the artificially occluded objects are shown in Figure 3. Although the training and test data are similar, the matching is not necessarily easy as the multimodal likelihood (see Figure 1) causes also the posterior to become multimodal.

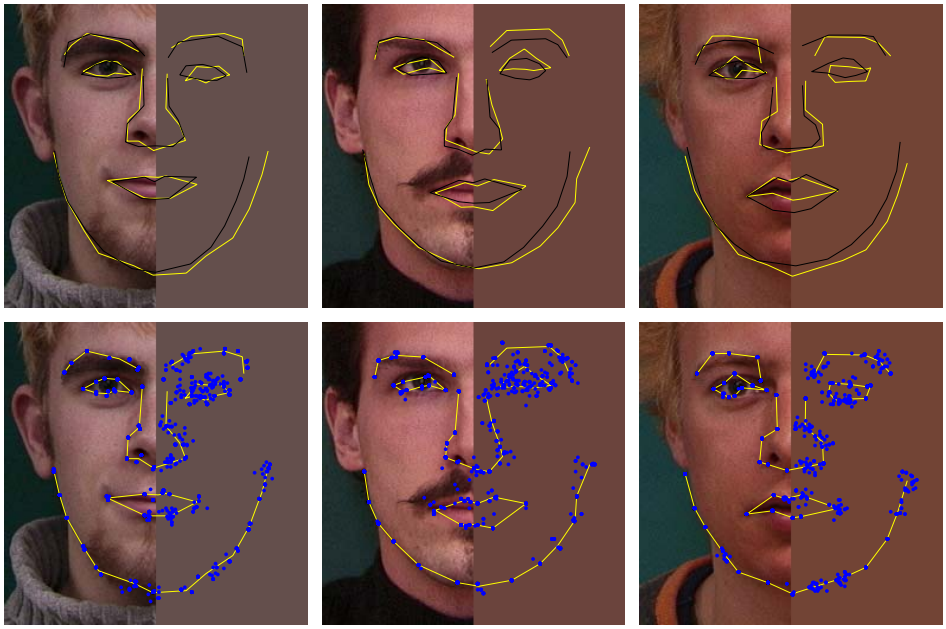

Figure 3: Matching results for images with simulated occlusions. In the top row, the light graphs are the sample means and the dark graphs the ground truth. In the bottom row, the light grids are again the sample means and the dots represent a part of the samples. Note how the variance of the samples is higher for most of the occluded features.

The matching error was assessed by measuring the point-to-point error (P2P) and the

point-to-curve error (P2C) (measuring the distance from the closest curve point) from the manual annotations. We measured the effect of the occlusion to the matching error of the visible features, the occluded features, as well as to the mean error of all features. To get a baseline comparison, the errors were also computed for unoccluded objects with and without the occlusion model. Furthermore, we compared the results to the AAM framework [2] implementation presented by Stegmann [13]. To decrease the effect of sampling variance on the results, the computations for our model were repeated 20 times and the mean error computed over all repetitions. The results are shown in Table 1. For the visible features, the increase in error is small, around 1 pixel (P2P)/0.5 pixels (P2C). There are two reasons for this increase: first, since fewer features are detected, there is less information available about their locations. Second, due to the randomization of the matching order, some visible features are bound to be matched after some occluded features, which hampers the matching performance because of the conditioning on the previously matched features. The matching error of the occluded features as well as the mean error of all the features are clearly higher, as can be expected, but especially the point-to-curve errors are still very reasonable. If all the features are visible, the occlusion model increases the matching error slightly due to the uncertainty about the visibility of the features.

The images with real occlusions were taken with a digital camera in uncontrolled office lighting conditions. The IMM-DTU images were used as the training data, with some results shown in Figure 4. Here we used a tighter prior than above due to the heads being smaller and the images being of poorer quality than the IMM-DTU images - final standard deviation was about 2 pixels. The results are promising: the IMM-DTU images (see Figures 1, 2 and 3) are *very* different from our images, and still the system performs well in most cases. For these images we had no ground truth available, and thus we can only demonstrate the results visually.

Table 1: Matching Results For Visible and Occluded Features

Error measurement	P2P error	P2C error
Simulated occlusions, proposed method:		
Visible features	6.96	3.50
Occluded features	11.3	6.89
All features	9.29	5.31
No occlusions, all features:		
Proposed method	6.22	3.30
Proposed method without occlusion model	5.57	2.78
Grayscale AAM	5.74	3.04
Color AAM	5.54	2.93

A drawback of any sampling-based scheme is computational complexity. Currently, with our unoptimized MATLAB implementation, the matching of a single image takes about 3 minutes on a regular Pentium IV PC.

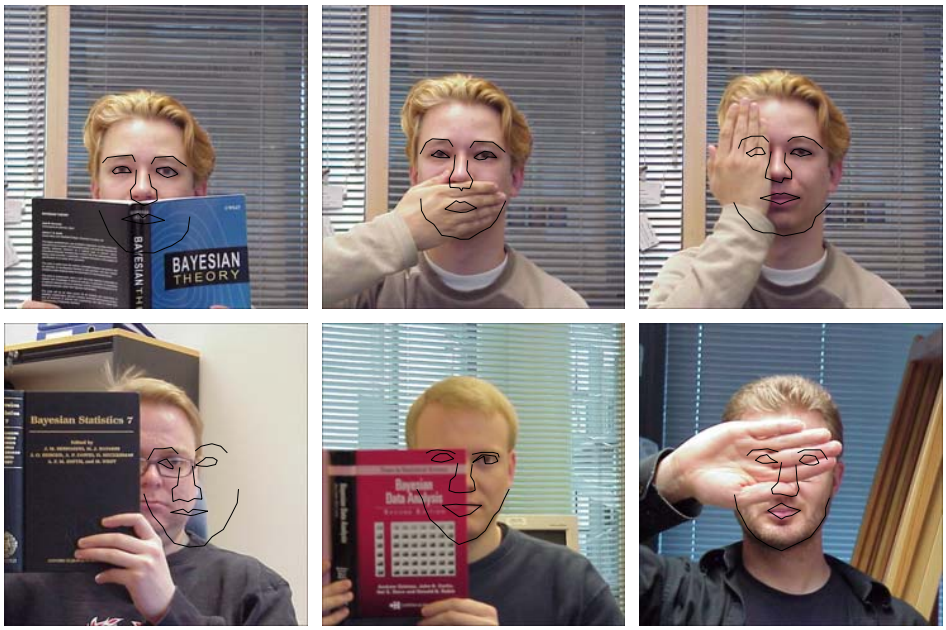

Figure 4: Matching results for images with real occlusions. The graphs show the sample means. The results are very good: only in the first image of the second row the matching has failed due to the likelihoods being low for almost all features. For example, the eyeglasses pass directly over the eye, which distorts the eye features.

7 Conclusion

We have presented a novel Bayesian occlusion model for a feature-based object matching system utilizing pixelwise likelihood computation, learned object models and sequential Monte Carlo sampling. The proposed system can handle complex features and object models, with good matching results even for heavily occluded objects.

Acknowledgments

The authors would like to thank Dr. Aki Vehtari for his help in the implementation of the samplers and both Dr. Vehtari and the anonymous reviewers for their helpful comments on the manuscript. Toni Tamminen gratefully acknowledges the support of his work by The Finnish Cultural Foundation.

References

[1] N. Chopin. A sequential particle filter method for static models. *Biometrika*, 89:539–552, 2002.

[2] T.F. Cootes, G.J. Edwards, and C.J. Taylor. Active appearance models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(6):681–685, 2001.

[3] J. G. Daugman. Complete discrete 2-d Gabor transforms by neural networks for image analysis and compression. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 36:1169–1179, 1988.

[4] A. Doucet, N. de Freitas, and N. Gordon, editors. *Sequential Monte Carlo Methods in Practice*. Springer, 2001.

[5] P.F. Felzenszwalb. Representation and detection of deformable shapes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2003.

[6] D. Fox, S. Thrun, W. Burgard, and F. Dellaert. Particle filters for mobile robot localization. In A. Doucet, N. de Freitas, and N. Gordon, editors, *Sequential Monte Carlo Methods in Practice*. Springer, 2001.

[7] A. Gelman, J. B. Carlin, H. S. Stern, and D. R. Rubin. *Bayesian Data Analysis*. Chapman & Hall, second edition, 2004.

[8] W.R. Gilks, S. Richardson, and D.J. Spiegelhalter, editors. *Markov Chain Monte Carlo in Practice*. Chapman & Hall, 1996.

[9] J.J. Koenderink and A.J. van Doorn. Representation of local geometry in the visual system. *Biological Cybernetics*, 55(6):367–375, 1987.

[10] F.-F. Li, R. Fergus, and P. Perona. A Bayesian approach to unsupervised one-shot learning of object categories. In *Proceedings of the International Conference on Computer Vision*, 2003.

[11] W.J. Rucklidge. Efficiently locating objects using the Haussdorf distance. *International Journal of Computer Vision*, 24(3):251–270, 1997.

[12] C. Steger. Occlusion, clutter, and illumination invariant object recognition. In *International Archives of Photogrammetry and Remote Sensing*, volume XXXIV, part 3A, 2002.

[13] M. B. Stegmann. Analysis and segmentation of face images using point annotations and linear subspace techniques. Technical report, Informatics and Mathematical Modelling, Technical University of Denmark, 2002.

[14] J. Sullivan, A. Blake, M. Isard, and J. MacCormick. Bayesian object localisation in images. *International Journal of Computer Vision*, 44(2):111–135, 2001.

[15] L. Wiskott, J.-M. Fellous, N. Krüger, and C. von der Malsburg. Face Recognition by Elastic Bunch Graph Matching. In L.C. Jain, U. Halici, I. Hayashi, and S.B. Lee, editors, *Intelligent Biometric Techniques in Fingerprint and Face Recognition*. CRC Press, 1999.

[16] Y. Zhong, A.K. Jain, and M.-P. Dubuisson-Jolly. Object tracking using deformable templates. *IEEE Transactions on Pattern Analysis and Machine Intellingence*, 22(5):544–549, 2000.