# Facial Landmark Localization using Robust Relationship Priors and Approximative Gibbs Sampling

Karsten Vogt, Oliver Müller and Jörn Ostermann

Institut für Informationsverarbeitung (TNT)
Leibniz Universität Hannover, Germany
{vogt, omueller, ostermann}@tnt.uni-hannover.de

**Abstract.** We tackle the facial landmark localization problem as an inference problem over a Markov Random Field. Efficient inference is implemented using Gibbs sampling with approximated full conditional distributions in a latent variable model. This approximation allows us to improve the runtime performance 1000-fold over classical formulations with no perceptible loss in accuracy. The exceptional robustness of our method is realized by utilizing a $L_1$-loss function and via our new robust shape model based on pairwise topological constraints. Compared with competing methods, our algorithm does not require any prior knowledge or initial guess about the location, scale or pose of the face.

## 1 Introduction

Accurate facial landmark localization algorithms play a vital role for many applications, such as biometric authentication [1] or human-machine-interfaces [2]. The goal of these algorithms is to estimate the pixel-coordinates of a configuration of predefined facial landmarks in an input image. Research has been mostly focused on three different approaches. First are methods based on a global appearance model [3]. Next are the part-based methods [4,5]. There, the individual landmarks are detected separately, while a shape model, typically based on a *point-distribution-model* (PDM), acts as a prior that constrains the space of valid configurations. The third class of methods are based on shape regression [6–9]. Regressors are trained to predict improved location estimates of all landmarks based on a previous guess. The relationship between landmarks is typically not explicitly modelled, but either learned from training data or enforced via regularization.

Past works usually focused on fast and accurate localization, while robustness was more of an afterthought. For example, methods that utilize an *Active Shape Model* (ASM) [10] require the approximate position and scale of the face to be known a-priori, as the landmark coordinates need to be aligned with a mean shape. Methods based on shape regression also require an initial guess. Arashloo et. al. [11] proposed a method to perform global optimization in a Markov Random Field (MRF) formulation of the landmark localization problem.

This approach is promising, but its implementation has two issues. First, the use of an ASM still retains the aforementioned shape alignment issues. Secondly, for each landmark only a small number of heuristically selected locations are ever considered. We wish to present a new landmark localization framework which is based on Bayesian principles that will solve the aforementioned problems and prioritizes robustness, while also achieving competitive performance and near real-time speed.

In this work, we tackle the facial landmark localization problem as an inference problem over an MRF. Efficient inference is implemented in the *Markov Chain Monte Carlo* MCMC framework, namely using Gibbs sampling. Our approach does not require a favorable initial estimate of the landmark locations. In fact, a completely random initialization will suffice. Our Gibbs sampler has the capability to rapidly cover the entire configuration space. We achieve large speed-ups over classical Gibbs sampling formulations by decomposing the full conditional distributions into sets of discrete latent variables. To solve the resulting sampling problem, we propose to approximate their probability distributions by exploiting the factorization of the posterior. We also propose a new PDM based shape model that is, in contrast to ASM and its variants, translation and scale invariant. This shape model consists of two components. The first component models the topology of the landmark configuration by imposing a set of simple pairwise relationship rules. The second one is inspired by *Shape-Indexed-Features* [12] and models the exact landmark locations in relation to nearby landmarks. Its main task is to fine-tune the results after the optimization has already mostly converged.

Our main contributions can be briefly stated as follows:

1. We present an approximation of the full conditional distributions that allow for speed-ups by a factor $> 1000$ over classical Gibbs sampling.
2. This approximation works in conjunction with a new translation and scale invariant shape model.
3. We always optimize over the full configuration space without requiring subsampling while also being extremely robust to bad initializations.

Section 2 introduces factor graphs. After formulating the landmark localization problem as an MRF inference problem, we present our new Gibbs sampling algorithm in Section 3 and our face model in Section 4. Section 5 ties everything together into a complete landmark localization framework. We evaluate our work in Section 6 and finish with conclusions in Section 7.

## 2 Graphical Models

The posterior probability of a facial landmark configuration $\boldsymbol{c} = (x_1, y_1, \ldots, x_L, y_L)$ with $L$ landmarks can be modeled as a factor graph $G = (V, F, E)$, where the set of vertices $V = \{v_1, \cdots, v_L\}$ represent the individual landmarks. The factors $F = \{f_1, \ldots, f_{|F|}\}$ define the relationships between vertices and are connected to them via undirected edges $E = \{e_1, \ldots, e_{|E|}\}$. A set of vertices $\mathcal{N}_f$ connected to the same factor $f \in F$ is called a *clique* and we will denote $\mathcal{F}_v$ as the set of factors that are connected to vertex $v$. We also define the probability of a configuration $\boldsymbol{c}$ conditioned on the observed image data $I$ as

$$p(\boldsymbol{c}\,|I(x,y)\,) \propto \prod_{f \in F} p_f^{\alpha_f}(\boldsymbol{c}(\mathcal{N}_f)\,|I\,)\ , \tag{1}$$

where $\boldsymbol{c}(\nu)$ are the coordinates for a subset of landmarks $\nu \subseteq V$, $p_f(\boldsymbol{c}(\mathcal{N}_f)\,|I\,)$ is the clique potential for a factor $f \in F$ and $\alpha_f$ is a tuning parameter that adjusts the influence of said factor.

Our objective is to find a landmark configuration which maximizes the posterior in Eq. (1) via inference on the graphical model $G$. Efficient inference on general graphical models is a notably complex problem. In the past, several different approaches have been proposed. The most successful ones have been *Gibbs sampling* [13], *belief propagation* [14] and *dual decomposition* [15]. While Gibbs sampling has been mostly succeeded by competing approaches in convergence speed, it still allows for a more natural handling of large clique sizes than either belief propagation or dual decomposition. Furthermore, all these methods struggle to perform well as the configuration space becomes larger. Assuming a full-HD image, each landmark can be situated in one of $1920 \times 1080$ different locations. The full configuration space is therefore comprised of $2.073.600^L$ different configurations. Efficient solutions can still be achieved either by subsampling the configuration space [11] or via particle-sampling [16]. Conceptually, both approaches achieve their runtime gains by considering only part of the full configuration space. This can be problematic, since the global optimum may not even be among the candidate configurations.

In this paper, we propose to solve the inference problem via *Gibbs sampling*. In contrast to competing methods, we will always consider the full configuration space. Our landmark detector will therefore be significantly more robust with regards to its initialization. Large speed-ups will be gained by introducing appropriate latent variables into the Gibbs sampling formulation.

## 3 Approximative Gibbs Sampling

Sampling based detectors first draw a representative random sample of configurations $\{\boldsymbol{c}_0, \ldots, \boldsymbol{c}_N\}$ from the posterior $p(\boldsymbol{c}\,|I\,)$. Different types of estimates can then be derived from this sample to find the solution that is best supported by the observed data. Gibbs sampling generates such a sample by sequentially generating new configurations $\boldsymbol{c}_{t+1}$ from $\boldsymbol{c}_t$ by sampling from each variable $\boldsymbol{c}(v_i)$ in turn, while keeping all other variables fixed. By exploiting the factorization of the posterior, we can simplify the full conditionals by discarding clique factors that are conditionally independent from the target variable:

$$\begin{aligned}\boldsymbol{c}(v_1) &\sim p(\boldsymbol{c}(v_1)\,|\,\boldsymbol{c}(V\setminus v_1), I\,) \propto \prod_{f\in\mathcal{F}_{v_1}} p_f^{\alpha_f}(\boldsymbol{c}(\mathcal{N}_f)\,|\,I\,)\\ &\vdots\\ \boldsymbol{c}(v_L) &\sim p(\boldsymbol{c}(v_L)\,|\,\boldsymbol{c}(V\setminus v_L), I\,) \propto \prod_{f\in\mathcal{F}_{v_L}} p_f^{\alpha_f}(\boldsymbol{c}(\mathcal{N}_f)\,|\,I\,)\ . \end{aligned} \tag{2}$$

At each Gibbs sampling step, we have to draw a variate from one of these discrete distributions. Even though there are well known algorithms to sample from arbitrary discrete distributions in constant time [17], these still require a linear time preprocessing step. As the configuration space becomes very large, the computational requirements of exact sampling can become prohibitive.

We solve this problem with a sampling strategy that can be best described as divide-and-conquer sampling. As shown in Fig. 1, we recursively subdivide the configuration space until we end up with elementary events. At each split, we have to sample from a latent variable $\phi_{v,i}$ with $M$ possible outcomes. Instead of sampling from one variable with $K$ outcomes, we sample from $\lceil \log_M(K)\rceil$ variables, each with only $M$ outcomes.

Next, we have to define the probability distributions of the latent variables. Since the landmark locations are inherently two-dimensional, sampling is actually performed with a quad-tree structure ($M = 4$). The probabilities for landmark $v$ to be in one of the four quadrants $Q_{i,j}$ ($j \in \{TL, TR, BL, BR\}$) can be directly stated as:

$$p_{\phi_{v,i}}(j) \propto \sum_{(x,y)\in Q_{i,j}} \prod_{f\in\mathcal{F}_v} p_f^{\alpha_f}(\boldsymbol{c}(\mathcal{N}_f)\,|\,I\,)\ . \tag{3}$$

This formulation requires us to evaluate the sum by computing the landmark location probability for each valid coordinate, which has a runtime complexity linear to the number of pixels in quadrant $Q_{i,j}$. If possible, we want to transform the problem such that the summation can be computed in constant time. Here is where our approximation comes into play. By upper-bounding the quadrant probabilities in Eq. (3) using the generalized *Hölder* inequality and renormalizing, we get the following approximated quadrant probabilities:

$$\tilde{p}_{\phi_{v,i}}(j) = \frac{1}{Z} \prod_{f\in\mathcal{F}_v} \left( \sum_{(x,y)\in Q_{i,j}} p_f^{\alpha_f\cdot|\mathcal{F}_v|}(\boldsymbol{c}(\mathcal{N}_f)\,|\,I\,) \right)^{1/|\mathcal{F}_v|}\ , \tag{4}$$

where $Z$ is a normalizing constant. This greatly simplifies the complexity of each sum and allows us to directly compute their result in constant time, for some well chosen families of factor distributions, independent of the size of the quadrant. Next, we will present clique potentials which are suitable for facial landmark detection and which also fulfill the required constant time complexity.

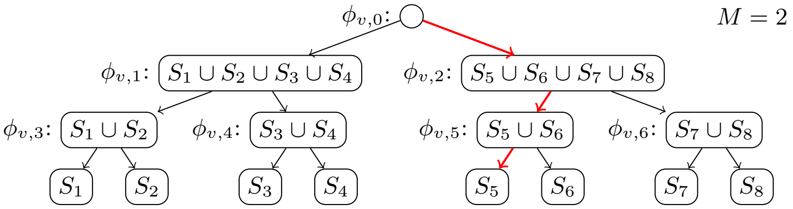

$\phi_{v,0}$:
$M = 2$
$\phi_{v,1}$: $S_1 \cup S_2 \cup S_3 \cup S_4$
$\phi_{v,2}$: $S_5 \cup S_6 \cup S_7 \cup S_8$
$\phi_{v,3}$: $S_1 \cup S_2$
$\phi_{v,4}$: $S_3 \cup S_4$
$\phi_{v,5}$: $S_5 \cup S_6$
$\phi_{v,6}$: $S_7 \cup S_8$
$S_1$
$S_2$
$S_3$
$S_4$
$S_5$
$S_6$
$S_7$
$S_8$

**Fig. 1.** Sampling in the latent variables formulation on a discrete variable $v$ with states $S_1, \ldots, S_8$. The red path shows an exemplary sampling path resulting in state $S_5$.

## 4 Face Model

### 4.1 Appearance Model – Unary Potentials

Unary potentials are only conditioned on the image data and describe the location probabilities for individual landmarks. Recent publications propose very diverse strategies to define such distributions. This broad range of approaches include *template matching* [18], MOSSE *filters* [4], *geometric blur features* [19] and HOG *features* [20]. In this paper, we chose to implement dense HOG *features* as described in [21]. They are invariant to changes in brightness, robust to changes in contrast, and robust to affine transformations by their very design. Location probabilities can be derived from these HOG features using any supervised classifier with soft outputs. We chose to use a multi-class linear SVM with calibrated outputs [22], as it is fairly robust to outliers and noise while also having a low amount of hyperparameters that need to be hand-tuned.

The summation over location probabilities in Eq. (4) can be efficiently computed in constant time by transforming it into a simple look-up operation. Since the unary potentials provide a single location probability for each landmark/pixel, independent of the location of neighboring landmarks, we can transfer the work to a preprocessing step, e.g., via summed-area-tables.

### 4.2 Shape Model – Higher Order Local Gaussian Potentials

The higher order potentials ($\geq$ 2) in our factor-graph model should describe the spatial relationship between different landmarks. We only draw new coordinates for one landmark at a time during each Gibbs sampling step, as all other landmarks remain fixed. Thus, we can directly model these relationships with bivariate probability distributions that are conditioned on the fixed coordinates. Due to the way our approximate sampling scheme is set up, and based on our own observations, these distributions should have the following properties:

1. To increase the expressiveness of the shape model, local shape components should be independently deformable.
2. If possible, we want to achieve invariance to translation and scale.
3. The sums in Eq. (4) must be evaluable in constant time.

Property 1 leads us to consider a local model that obeys the Markov property, i.e., each landmark $v_i$ is only dependent on the positions of its direct spatial neighbors. Property 2 strongly favors models that operate in an appropriately chosen local coordinate system, which is constructed from these neighbors. Of course, the local coordinates would not be very informative if the neighborhood system is allowed to change during the course of the simulation. Therefore, the neighborhood system will first be extracted from a mean face shape $\bar{S}$, which is estimated from a training set via *Generalized Procrustes Analysis*. After constructing the *Delaunay* triangulation of all landmarks $V \setminus v_i$ on $\bar{S}$, the three neighbors $n_1(v_i)$, $n_2(v_i)$ and $n_3(v_i)$ are simply the vertices of the encompassing triangle of $v_i$. We can now construct a non-orthogonal local basis for $v_i$ from any landmark configuration as shown in Fig. 2(a). We first select $\boldsymbol{o} = \boldsymbol{c}(n_1(v_i))$ as the origin of this coordinate system and $\boldsymbol{x} = \boldsymbol{c}(n_2(v_i)) - \boldsymbol{o}, \boldsymbol{y} = \boldsymbol{c}(n_3(v_i)) - \boldsymbol{o}$ as its basis vectors. Transforming a coordinate vector $\boldsymbol{c}(v_i)$ from the global basis to the local basis can be achieved as follows: $\boldsymbol{c}_{local}(v_i) = A \cdot (\boldsymbol{c}(v_i) - \boldsymbol{o})$ , where $A = \left[\frac{\boldsymbol{x}}{\boldsymbol{x}^T\boldsymbol{x}} \; \frac{\boldsymbol{y}}{\boldsymbol{y}^T\boldsymbol{y}}\right]^T$ . The bivariate location distribution for each landmark can now be defined as a Gaussian distribution parametrized in the local coordinate system.

$$
\begin{aligned}
p_f(\boldsymbol{c}(\mathcal{N}_f)\,|I\,) &\propto p(\boldsymbol{c}(v_i)\,|\boldsymbol{c}(n_1(v_i)), \boldsymbol{c}(n_2(v_i)), \boldsymbol{c}(n_3(v_i))\,) && (5)\\
\boldsymbol{c}_{local}(v_i) &\sim \mathcal{N}(\mu_{local}, \Sigma_{local}) && (6)\\
\Leftrightarrow \boldsymbol{c}(v_i) &\sim \mathcal{N}(A^{-1}\mu_{local} + \boldsymbol{o}, A^{-1}\Sigma_{local}A^{-T})\ . && (7)
\end{aligned}
$$

The local distribution parameters $\mu_{local}$ and $\Sigma_{local}$ can be estimated from the training set. Since the transformation between the local and global bases is simply a linear relationship, we can also reproject the local distribution back into the global space (Eq. (7)). For landmarks situated on the convex hull of $\bar{S}$, we will have to employ a different procedure to select an appropriate local basis. It turned out that the robustness of the location estimate is more important than its accuracy. For these landmarks we will therefore select $K$ uniformly distributed bases at random, each inducing an independent estimate of their location. The final distribution for $\boldsymbol{c}(v_i)$ represents the consent between all $K$ estimates and can than be derived by multiplying their normal distributions in the global space, which again produces a single bivariate normal distribution.

The product of Gaussians is itself proportional to a Gaussian distribution [23]. As required for Property 3, the sum in Eq. (4) can therefore be evaluated using the bivariate cumulative distribution function (CDF) of the Gaussian distribution in Eq. (7), e.g. using the algorithm described in [24].

### 4.3 Shape Model – Rule-Based Binary Potentials

The local Gaussian relationship model is prone to slow convergence and can be unstable if the initial solution is not chosen well. We tackle this problem by augmenting our graphical model with additional robust binary factors. While the local Gaussian factors model the geometry of the shape, these binary factors should only model its topology. Each binary clique represents a simple relationship rule. These rules may be of the form $r(f, v_i, v_j) \in \{\text{is-left-of}, \text{is-right-of},$

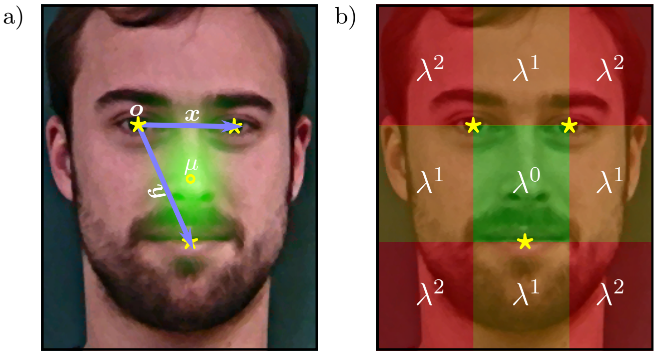


**Fig. 2.** Higher-order potentials for the nose landmark. (a) Gaussian distribution with mean $\mu$ and its local coordinate system and (b) penalty regions for the rule-based prior with penalty term $\lambda$.

is-above, is-below}. The clique distribution of a single rule is then defined as

$$p_f(\boldsymbol{c}(v_i), \boldsymbol{c}(v_j)) \propto p_f(\boldsymbol{c}(v_i)\,|\,\boldsymbol{c}(v_j)\,) = \lambda^{1_{(\text{rule } r(f,v_i,v_j) \text{ is violated})}} , \tag{8}$$

where $\lambda$ is a user-adjustable penalty term and $1_A$ is the indicator function. Multiple rules operating on the same landmark $v_i$ are statistically independent and their conditional distribution may be jointly calculated by counting how many rules are being violated by the proposed configuration. The higher-order clique distribution over all landmarks is then defined as

$$p(\boldsymbol{c}(v_i)\,|\,\boldsymbol{c}(V \setminus v_i)\,) = \lambda^{\sum_{f \in \mathcal{F}_{v_i}} 1_{(\text{rule } r(f,v_i,v_j) \text{ is violated})}} . \tag{9}$$

Allowing only the specified types of rules, the probability distribution in Eq. (9) is always piecewise-constant and axis-aligned. Evaluation of the sum for the quadrant probabilities in Eq. (4) can therefore be implemented such that the runtime complexity is invariant to the image size and only depends on the number of rules. This relationship model has the advantage of being fairly simple and robust, yet also invariant to translation and scale. Robustness to rotations can be improved by augmenting the training set with slightly rotated versions of the input faces. We automatically select rules by including all relationship rules that hold true for at least 95% of all images in the training dataset.

## 5 Landmark Localization Algorithm

Our facial landmark localization algorithm uses the Gibbs sampling scheme (Section 3) as its core component. Yet, there are a few details that could not be covered in the previous sections. Algorithm 1 presents our algorithm AGS (*Approximative Gibbs Sampling*) in pseudo code. We explain the individual components of our algorithm step-by-step:

**Algorithm 1** AGS landmark localization algorithm.

```
function LocalizeLandmarks(Image $I_{\text{RGB}}$)
Parameters:
    Sample-Chains $C$, Burn-In Samples $B$, Sample-Size $N$, Loss-Function $\mathcal{L}$
Preprocessing:
    $I_{\text{YUV}} \leftarrow$ RGB2YUV($I_{\text{RGB}}$)
    for $pose \in \{ProfileLeft, Frontal, ProfileRight\}$ do
        $U_{pose} \leftarrow$ UnaryPotentials($I_{\text{YUV}}$, $pose$)
        $T_{pose} \leftarrow$ ConstructSummedAreaTable($U_{pose}$)
Sampling:
    $S \leftarrow$ Empty List                                        ▷ Initialize list of samples
    for each Sample-Chain $\in [1 \dots C]$ do
        for $pose \in \{ProfileLeft, Frontal, ProfileRight\}$ do
            $\boldsymbol{c}_{pose,0} \leftarrow$ Random Configuration
            for $i = 1 \dots {}^{B}/_{2}$ do $\boldsymbol{c}_{pose,i} \leftarrow$ GibbsSampling($\boldsymbol{c}_{pose,i-1}$, $T_{pose}$, $pose$)
        $\theta \leftarrow \arg\max_{\theta} p(\boldsymbol{c}_{\theta,B/2}\,|\,I)$        ▷ Select $pose$ $\theta$ which maximizes the posterior
        for $i = {}^{B}/_{2} \dots B + N$ do
            $\boldsymbol{c}_{\theta,i} \leftarrow$ GibbsSampling($\boldsymbol{c}_{\theta,i-1}$, $T_{\theta}$, $\theta$)
            if $i > B$ then Append $\boldsymbol{c}_{\theta,i}$ to $S$
Estimation:
    switch $\mathcal{L}$ do
        case MAP: return $\arg\max_{\mathbf{c} \in S} p(\boldsymbol{c}\,|\,I)$
        case $L_1$-loss: return MEDIAN($S$)
        case $L_2$-loss: return MEAN($S$)
```

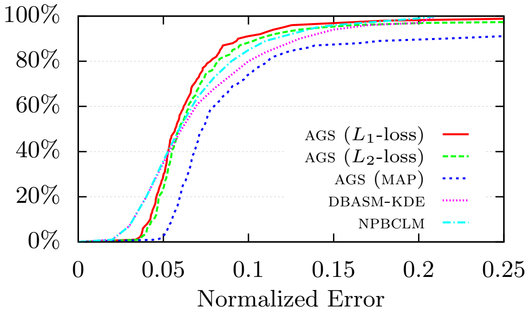
Pose is discretized into three possible states: frontal, left looking and right looking, and for each of these poses a learned shape model is available. During the preprocessing step, we will first precompute the unary potentials. The HOG-features are computed separately for each color channel in the YUV color space. We accelerate the computation of the quadrant probabilities in Eq. (4) by storing the unary potentials in a summed-area-table. The preprocessing step of the algorithm is actually the most performance critical one, because we have to evaluate the location probabilities for each landmark and pixel coordinate at the original image resolution. Therefore it is paramount to use optimized implementations for the dense HOG-feature extraction and classification. We then create multiple independent sampling chains, each initialized with a random landmark configuration. Pose estimation is handled in a very straight-forward manner. For each chain, we simply try all three pose states and advance the sampling chain for a few iterations using our approximate Gibbs sampler. The pose that results in the maximum a-posteriori configuration will then be selected for this chain. Following this, we draw the remaining samples from the sampling chain.

## 6 Evaluation

Here, we evaluate our landmark detector with respect to its landmark localization error and convergence properties. We compare our results with the DBASM algorithm [4] and the npBCLM algorithm [25], since both are recent part-based landmark detector grounded in Bayesian methodology. Unless otherwise noted, all experiments are performed with 8 independent sampling chains and a sample-size of 500 per chain, of which the first 100 burn-in samples are discarded. Other

**Table 1.** Parameters for the AGS landmark detector.

| | | | | | |
|---|---|---|---|---|---|
| HOG block_size | $36 \times 36$ | HOG normalization | $L_2$ | $\alpha_{\text{unary}}$ | 1.0 |
| HOG cell_size | $9 \times 9$ | SVM C | 1.0 | $\alpha_{\text{Gaussian}}$ | 0.025 |
| HOG num_bins | 8 | rule-based penalty $\lambda$ | 0.1 | $\alpha_{\text{rule-based}}$ | 0.25 |



| | % < 0.1 Norm. Error | % < 7.5 RMS Error |
|---|---|---|
| AGS (MAP) | 74.1 | 46.3 |
| AGS ($L_2$-loss) | 88.5 | 65.0 |
| AGS ($L_1$-loss) | **91.4** | **70.4** |
| DBASM-KDE | 79.5 | 64.6 |
| npBCLM | 84.9 | - |

**Fig. 3.** Cross-validation results for the IMM dataset. The graph shows percentile plots over normalized landmark localization errors. The table compares the performance of different algorithms and presents the proportion of images in the dataset, for which the localization error falls below a predefined threshold (higher is better).

parameters were tuned empirically and set as shown in Table 1. Each chain is always initialized with a random landmark configuration.

**IMM Dataset:** The IMM dataset [26] contains 240 annotated images of 40 different subjects with a resolution of 640 × 480 pixels. The annotations for each face image include 58 different landmarks and the sex of the subject. All face images were captured indoors under studio condition but vary in pose, size, facial expression and lighting.

**Localization error:** The automatically detected landmark locations should be as close as possible to a manually created ground-truth. Additionally, the detector should also generalize well to unseen faces. To this end, we perform a 40-fold cross-validation by partitioning the dataset by subject. Errors are measured as the interocular distance normalized error averaged over all images in the dataset. Fig. 3 shows the results for our AGS algorithm with three different loss functions. The MAP estimator simply selects the single best landmark configuration that achieves the highest posterior probability from all sampling chains. The Bayesian estimator with $L_2$-loss assumes a normally distributed localization error, while the Bayesian estimator with $L_1$-loss assumes longer tails for the posterior distribution and is therefore more robust to outliers. As can be clearly seen, a simple MAP estimate is too noisy to get satisfactory results. In comparison, both $L_2$ and $L_1$ loss functions will always generate significantly improved location estimates. Our method also improves on the results of the DBASM algorithm using a kernel density estimate and the npBCLM algorithm. Fig. 4 shows some typical results for this dataset.

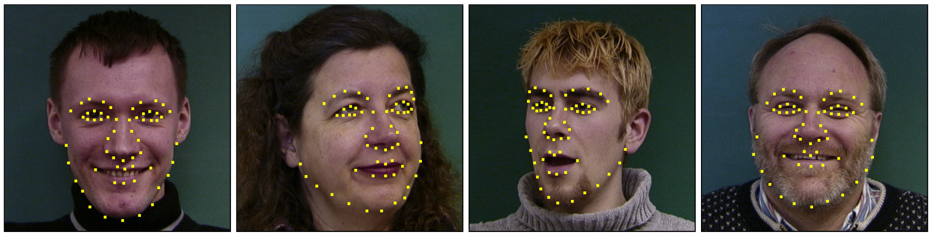


**Fig. 4.** Example results for the IMM dataset.

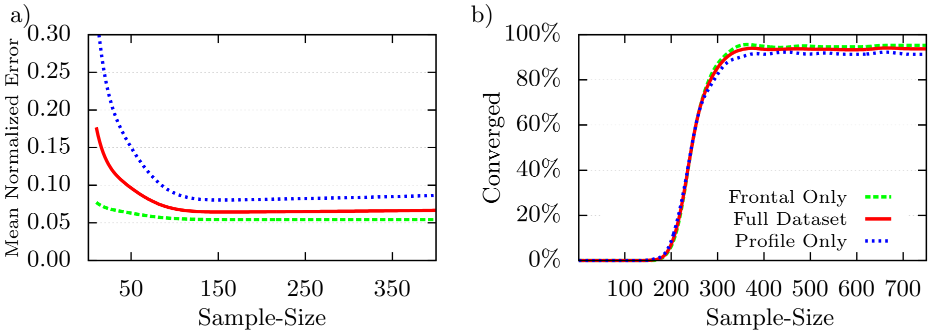


**Fig. 5.** Convergence results for the IMM dataset. (a) Mean normalized error over sample-size and (b) percent of converged images over sample-size.

**Convergence:** Convergence can be measured in multiple ways. Our method has fast convergence in both a theoretical and a practical sense. Fig. 5(a) shows the average normalized landmark location error as a function of the sample size. We show that the sampling chains typically show fast mixing behavior by calculating the multivariate scale reduction factor $\hat{R}^p$ as in [27]. For mixing chains, $\hat{R}^p$ should start to approach 1.0 with increasing sample sizes. Fig. 5(b) shows the percentage of input images for which the sampling converged ($\hat{R}^p < 1.2$) within a specified sampling size. As can be seen in Fig. 5(a), less than 100 iterations will usually suffice to get close to optimal results. Frontal faces show slightly lower estimation errors than profile views, but on average the results are . Mixing behavior is almost impeccable. For more than 95% of the dataset, less than 300 iterations are sufficient to demonstrate convergence to the target distribution. Only a few particularly uncommon facial expressions or poses exhibit slow convergence.

**Runtime Performance:** We give the runtime for all three phases of our algorithm separately in table 2. The preprocessing step includes the computation of HOG features, pixelwise evaluation of the appearance model and preparation of the integral images. Sampling speed is shown for our approximative sampling scheme and for exact discrete Gibbs sampling. The measurements were averaged over a large number of draws. Currently, the major bottleneck of our algorithm is the precomputation of the appearance model. In case a high image throughput

**Table 2.** Runtime performance measurements. The experiments were done on a *Intel(R) Xeon(R) E5-2690* CPU at 3 *GHz* using 8 cores.

| Preprocessing | | Sampling | | Estimation |
|---|---|---|---|---|
| HOG Features | Unary Potentials | Approx. Sampling | Exact Sampling | |
| $270\,ms$ | $3000\,ms$ | $0.4\,ms/\text{draw}$ | $633\,ms/\text{draw}$ | $\ll 1\,ms$ |

is required, we suggest to pipeline the three phases of our localization algorithm. The preprocessing phase could than be distributed over multiple networked computers. The runtime improvements due to our approximation scheme are usually on the order of a 1600-fold reduction in sampling time for a $640 \times 480$ pixel image. Larger images will of course amplify the gain. Given the long compute time for exact Gibbs sampling, we do not present cross-validation results for this method.

## 7 Conclusion

This work presents a facial landmark localization algorithm based on Gibbs sampling with approximated full conditional distributions. Compared with competing methods, the presented algorithm does not require an initial guess and improves on their localization errors. A new robust shape model allows for translation and scale invariant landmark localization while generally achieving fast convergence for a variety of poses and facial expressions.

## References

1. Jain, A.K., Ross, A., Prabhakar, S.: An introduction to biometric recognition. TCSVT **14** (2004) 4–20
2. Mulligan, J.B.: A software-based eye tracking system for the study of air-traffic displays. In: ETRA, ACM (2002) 69–76
3. Cootes, T.F., Edwards, G.J., Taylor, C.J.: Active appearance models. TPAMI **23** (2001) 681–685
4. Martins, P., Caseiro, R., Henriques, J.F., Batista, J.: Discriminative bayesian active shape models. In: ECCV. Springer (2012) 57–70
5. Zhou, F., Brandt, J., Lin, Z.: Exemplar-based graph matching for robust facial landmark localization. In: ICCV, IEEE (2013) 1025–1032
6. Dollár, P., Welinder, P., Perona, P.: Cascaded pose regression. In: CVPR, IEEE (2010) 1078–1085
7. Dantone, M., Gall, J., Fanelli, G., Van Gool, L.: Real-time facial feature detection using conditional regression forests. In: CVPR, IEEE (2012) 2578–2585
8. Cao, X., Wei, Y., Wen, F., Sun, J.: Face alignment by explicit shape regression. International Journal of Computer Vision **107** (2014) 177–190
9. Ren, S., Cao, X., Wei, Y., Sun, J.: Face alignment at 3000 fps via regressing local binary features. In: CVPR, IEEE (2014) 1685–1692
10. Cootes, T.F., Taylor, C.J., Cooper, D.H., Graham, J.: Active shape models-their training and application. Computer vision and image understanding **61** (1995) 38–59

11. Arashloo, S.R., Kittler, J., Christmas, W.J.: Facial feature localization using graph matching with higher order statistical shape priors and global optimization. In: BTAS, IEEE (2010) 1–8
12. Burgos-Artizzu, X.P., Perona, P., Dollár, P.: Robust face landmark estimation under occlusion. In: ICCV, IEEE (2013) 1513–1520
13. Walsh, B.: Markov chain monte carlo and gibbs sampling. (2004)
14. Bishop, C.M., et al.: Pattern recognition and machine learning. Volume 4. Springer New York (2006)
15. Komodakis, N., Paragios, N., Tziritas, G.: Mrf energy minimization and beyond via dual decomposition. TPAMI **33** (2011) 531–552
16. Müller, O., Yang, M.Y., Rosenhahn, B.: Slice sampling particle belief propagation. In: ICCV, IEEE (2013) 1129–1136
17. Walker, A.J.: An efficient method for generating discrete random variables with general distributions. TOMS **3** (1977) 253–256
18. Yuille, A.L., Hallinan, P.W., Cohen, D.S.: Feature extraction from faces using deformable templates. International journal of computer vision **8** (1992) 99–111
19. Berg, A.C., Malik, J.: Geometric blur for template matching. In: CVPR, IEEE (2001) 607–614
20. Albiol, A., Monzo, D., Martin, A., Sastre, J., Albiol, A.: Face recognition using hog–ebgm. Pattern Recognition Letters **29** (2008) 1537–1543
21. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: CVPR, IEEE (2005) 886–893
22. Fan, R.E., Chang, K.W., Hsieh, C.J., Wang, X.R., Lin, C.J.: Liblinear: A library for large linear classification. The Journal of Machine Learning Research **9** (2008) 1871–1874
23. Bromiley, P.: Products and convolutions of gaussian distributions. Medical School, Univ. Manchester, Manchester, UK, Tech. Rep **3** (2003)
24. Genz, A.: Numerical computation of rectangular bivariate and trivariate normal and t probabilities. Statistics and Computing **14** (2004) 251–260
25. Martins, P., Caseiro, R., Batista, J.: Non-parametric bayesian constrained local models. In: CVPR, IEEE (2014) 1797–1804
26. Nordstrøm, M.M., Larsen, M., Sierakowski, J., Stegmann, M.B.: The IMM face database - an annotated dataset of 240 face images. Technical report, Informatics and Mathematical Modelling, Technical University of Denmark, DTU, Richard Petersens Plads, Building 321, DK-2800 Kgs. Lyngby (2004)
27. Brooks, S.P., Gelman, A.: General methods for monitoring convergence of iterative simulations. Journal of computational and graphical statistics **7** (1998) 434–455