

Occlusion Coherence: Detecting and Localizing Occluded Faces

Golnaz Ghiasi and Charless C. Fowlkes

Abstract—The presence of occluders significantly impacts object recognition accuracy. However, occlusion is typically treated as an unstructured source of noise and explicit models for occluders have lagged behind those for object appearance and shape. In this paper we describe a hierarchical deformable part model for face detection and keypoint localization that explicitly models part occlusion. The proposed model structure makes it possible to augment positive training data with large numbers of synthetically occluded instances. This allows us to easily incorporate the statistics of occlusion patterns in a discriminatively trained model. We test the model on several benchmarks for keypoint localization and detection including challenging data sets featuring significant occlusion. We find that the addition of an explicit model of occlusion yields a system that outperforms existing approaches in keypoint localization accuracy and detection performance.

Index Terms—Object Recognition, Face Detection, Occlusion, Deformable Part Model

arXiv:1506.08347v1 [cs.CV] 28 Jun 2015

1 INTRODUCTION

ACCURATE localization of facial keypoints provides an important building block for many applications including identification [1] and analysis of facial expressions [2]. Significant progress has been made in this task, aided in part by the fact that faces have less intra-category shape variation and limited articulation compared to other object categories of interest. However, feature point localization tends to break down when applied to faces in real scenes where other objects in the scene (hair, sunglasses, other people) are likely to occlude parts of the face. Fig. 1(a) depicts the output of a deformable part model [3] where the presence of occluders distorts the final alignment of the model.

A standard approach to handling occlusion in part-based models is to compete part feature scores against a generic background model or fixed threshold (as in Fig. 1(b)). However, setting such thresholds is fraught with difficulty since it is hard to distinguish between parts that are present but simply hard to detect (e.g., due to unusual lighting) and those which are genuinely hidden behind another object.

Treating occlusions as an unstructured source of noise ignores a key aspect of the problem, namely that occlusions are induced by other objects and surfaces in the scene and hence should exhibit **occlusion coherence**. For example, it would seem very unlikely that every-other keypoint along an object contour would happen to be occluded. Yet many occlusion models make strong independence assumptions about occlusion, making it difficult to distinguish *a priori* likely from unlikely patterns. Ultimately, an occluder should not be inferred simply by the lack of evidence for object features, but rather by positive evidence for the



Fig. 1: Occlusion impacts part localization performance. In panel (a) the output of a deformable part model [3] is distorted by the presence of occluders, disrupting localization even for parts that are far from the site of occlusion. (b) Introducing independent occlusion of each part results in better alignment but occlusion is treated as an outlier process and prediction of occlusion state is inaccurate. (c) The output of our hierarchical part model, which explicitly models likely patterns of occlusion, shows improved localization as well as accurate prediction of which keypoints are occluded.

- G. Ghiasi and C. Fowlkes are with the Department of Computer Science, University of California at Irvine, Irvine, CA 92697. E-mail: {gghiasi,fowlkes}@ics.uci.edu
- This work was supported by NSF grants IIS-1253538, DBI-1262547

occluding object that **explains away** the lack of object features.

The contribution of this paper is an efficient hierarchical

deformable part model that encodes these principles for modeling occlusion and achieves state-of-the-art performance on benchmarks for occluded face localization and detection (depicted in Fig. 1(c)). Building on our previously published results [4], we model the face by an arrangement of parts, each of which is in turn composed of local keypoint features. This two-layer model provides a compact, discriminative representation for the appearance and deformations of parts. It also captures the correlation in shapes and occlusion patterns of neighboring parts (e.g., if the chin is occluded it would seem more likely the bottom half of the mouth is also occluded). In addition to representing the face shape, each part has an associated occlusion state chosen from a small set of possible occlusion patterns, enforcing coherence across neighboring keypoints and providing a sparse representation of the occluder shape where it intersects the part. We describe the details of this model in Section 3.

Specifying training data from which to learn feasible occlusion patterns comes with an additional set of difficulties. Practically speaking, existing datasets have focused primarily on fully visible faces. Moreover, it seems unlikely that any reasonable sized set of training images would serve to densely probe the space of possible occlusions. Beyond certain weak contextual constraints, the location and identity of the occluder itself are arbitrary and largely independent of the occluded object. To overcome this difficulty of training data, we propose a unique approach for generating synthetically occluded positive training examples. By exploiting the structural assumptions built into our model, we are able to include such examples as “virtual training data” without explicitly synthesizing new images. This in turn leads to an interesting formulation of discriminative training using a loss function that depends on the latent occlusion state of the parts for negative training examples which we describe in Section 4.

We carry out an extensive analysis of this model performance in terms of keypoint localization, occlusion prediction and detection accuracy. The model we describe achieves high-quality keypoint localization accuracy, comparable to pose regression, while being more robust to initialization and occlusions (Section 5.1). We also find that the prediction of which keypoints are occluded is improved over a simple independent occlusion model (Section 5.2). Unlike keypoint regression methods, our model also functions as a detector and achieves good performance on standard face detection benchmarks such as FDDB [5]. Finally, to illustrate the impact of occlusion on existing detection models, we evaluate performance on a new face detection dataset that contains significant numbers of partially occluded faces (Section 5.3).

2 RELATED WORK

Face Detection and Localization: There is a long history of face detection in the computer vision literature. Classic approaches to 2D alignment include Deformable Templates [6], Active Appearance Models (AAMs) [7], [8], [9] and

elastic graph matching [10]. Alignment with full 3D models provides even richer information at the cost of additional computation [11], [1]. A key difficulty in most of these approaches is the dependence on iterative and local search techniques for optimizing model alignment with a query image. This typically results in high computational cost and the concern that local minima may undermine system performance.

A more recent family of approaches makes use of constrained local models that first detect candidate local features and then enforce constraints between parts [12]. Training regressors that learn to predict keypoint locations from both appearance and spatial context provided by other detector responses has also shown good performance [13], [14], [15], [16], [17]. A key advantage is that such pose-regression models can be trained layer-wise in a discriminative fashion and thus sidestep the optimization problems of global model alignment as well as providing fast, feed-forward performance at test time.

Our model is most closely related to the work of [3] which applies discriminatively trained deformable part models (DPM) [18] to face analysis. This offers an intermediate between the extremes of model alignment and keypoint regression, by utilizing mixtures of simplified shape models that make efficient global optimization of part placements feasible while exploiting discriminative training criteria. Similar to [19], we use local part and keypoint mixtures to encode richer multi-modal shape distributions. We extend this line of work by adding hierarchical structure and explicit occlusion to the model. We introduce intermediate part nodes that do not have an associated “root template” but instead serve to encode an intermediate representation of occlusion and shape state. The notion of hierarchical part models has been explored extensively as a tool for compositional representation and parameter sharing (see e.g., [20], [21]). While the intermediate state represented in such models can often be formally encoded in by non-hierarchical models with expanded state spaces and tied parameters, our experiments show that the particular choice of model structure proves essential for efficient representation and inference.

Occlusion Modeling: Modeling occlusion is a natural fit for recognition systems with an explicit representation of parts. Work on generative constellation models [22], [23] learned parameters of a full joint distribution over the probability of part occlusion and relied on brute force enumeration for inference, a strategy that doesn’t scale to large numbers of keypoints. More commonly, part occlusions are treated independently which makes computation and representation more efficient. For example, the supervised detection model of [24] associates with each part a binary variable indicating occlusion and learns a corresponding appearance template for the occluded state.

The authors of [21] impose a more structured distribution on the possible occlusion patterns by specifying grammar that generates a person detector as a variable length vertical chain of parts terminated by an occluder template, while

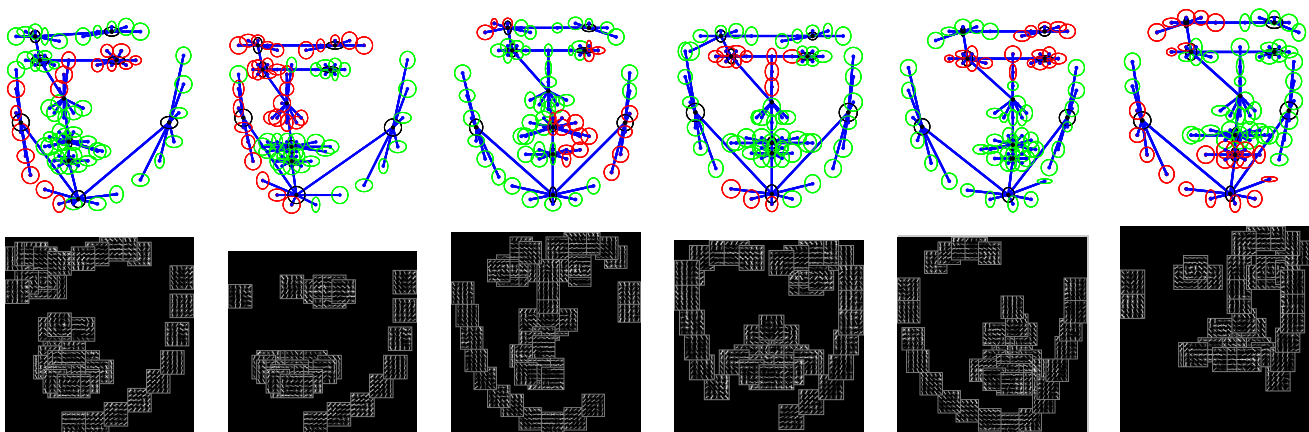


Fig. 2: Our model consists of a tree of parts (black circles) each of which is connected to a set of keypoints (green or red) in a star topology. The examples here show templates corresponding to different choices of part shape and occlusion patterns. Red indicate occluded keypoints. Shape parameters are independent of occlusion state. Keypoint appearance is modeled with a small HOG template (2nd row) and occluded keypoints are constrained to have an appearance template fixed to 0. Note how the model produces a wide range of plausible shape configurations and occlusion patterns.

[25] allows “flexible compositions” which correspond to occlusion patterns that leave visible a connected subgraph of the original tree-structure part model. Our approach provides a stronger model than full independence, capturing correlations between occlusions of non-neighboring keypoints. Unlike the grammar-based approach, occlusion patterns are not specified structurally but instead learned from data and encoded in the model weights.

Pose regression approaches have also been adapted to incorporate explicit occlusion modeling. For example, the face model of [26] uses a robust m-estimator which serves to truncate part responses that fall below a certain threshold. In our experiments, we compare our results to the recent work of [15] which uses occlusion annotations when training a cascade of regressors where each layer predicts both part locations and occlusion states.

3 HIERARCHICAL PART MODEL

In this section we develop a hierarchical part model that simultaneously captures face appearance, shape and occlusion. Fig. 2 shows a graphical depiction of the model structure. The model has two layers: the face consists of a collection of parts (nose, eyes, lips) each of which is in turn composed of a number of keypoints that specify local edge features making up the part. Keypoints are connected to their parent part nodes with a star topology while the connections between parts forms a tree. In addition to location, each part takes one of a discrete set of shape states (corresponding to different facial shapes or expressions) and occlusion states (corresponding to different patterns of visibility). The model topology which groups facial features into parts was specified by hand while the shape and occlusion patterns are learned automatically from training data (see Section 4). This model, which we term a hierarchical part model (HPM) is a close cousin of the deformable part model (DPM) of [18] and the flexible part model (FMP) of [3]. It differs in the addition of part nodes that model shape

but don’t include any “root filter” appearance term, and by the use of mixtures to model occlusion patterns for each part. In this section we introduce some formal notation to describe the model and some important algorithmic details for performing efficient message passing during inference.

3.1 Model Structure

Let l, s, o denote the hypothesized locations, shape and occlusion of N parts and keypoints describing the face. Locations $l \in \mathbb{R}^{2N}$ range over the whole image domain and $o \in \mathcal{O}_1 \times \mathcal{O}_2 \dots \times \mathcal{O}_N$ indicates the occlusion states of parts and keypoints. The shape $s \in \mathcal{S}_1 \times \mathcal{S}_2 \dots \times \mathcal{S}_N$ selects one of a discrete set of shape mixture components for each part. We define a tree structured scoring function by:

$$Q(l, s, o|I) = \sum_i \phi_i(l_i, s_i, o_i|I) \quad (1)$$

$$+ \sum_i \sum_{j \in \text{child}(i)} \psi_{ij}(l_i, l_j, s_i, s_j) + b_{ij}(s_i, s_j, o_i, o_j)$$

where the potential ϕ scores the consistency of the local image appearance around location l_i , ψ is a quadratic shape deformation penalty, and b is a co-occurrence bias.

The first (unary) term scores the appearance evidence. We linearly parameterize the unary appearance term with filter weights $w_i^{s_i}$ that depend on the discrete shape mixture selected

$$\phi_i(l_i, s_i, o_i|I) = w_i^{s_i} \cdot \phi(l_i, o_i|I)$$

Appearance templates are only associated with the leaves (keypoints) in the model so the unary term only sums over those leaf nodes. The occlusion variables o_i for the keypoints are binary, corresponding to either occluded or visible. If the i th keypoint is unoccluded, the appearance feature ϕ is given by a HOG [27] feature extracted at location l_i , otherwise the feature is set to 0. This is natural on theoretical grounds since the appearance of the occluder

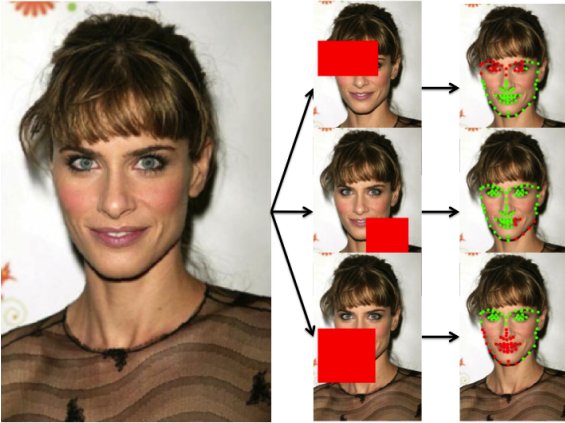


Fig. 3: Virtual positive examples are generated synthetically by starting with a fully visible training example and sampling random coherent occlusion patterns.

is arbitrary and hence indistinguishable from background based on its local appearance. Empirically we have found that unconstrained occluder templates learned with sufficiently varied data do in fact have very small norms, further justifying this choice [28].

The second (pairwise) term in Eq. 1 scores the placement part j based on its location relative to its parent i and the shape mixtures of the child and parent. We model this with a linearly parameterized function:

$$\psi_{ij}(l_i, l_j, s_i, s_j) = w_{ij}^{s_i, s_j} \cdot \psi(l_i - l_j)$$

where the feature ψ includes the x and y displacements and their cross-terms, allowing the weights w_{ij} to encode a standard quadratic “spring”. We assume that the shape of the parts is independent of any occluder so the spring weights do not depend on the occlusion states.¹ The pairwise parameter b_{ij} encodes a bias of particular occlusion patterns and shapes to co-occur. Formally, each keypoint has the same number of occlusion states and shape mixtures as its parent part, but we fix the bias parameters between the part and its constituent keypoints to impose a hard constraint that the mixture assignments are compatible.

3.2 Efficient Message Passing

The model above can be made formally equivalent to the FMP model used in [19] by introducing local mixture variables that live in the cross-product space of o_i and s_i . However, this reduction fails to exploit the structure of the occlusion model. This is particularly important due to the large size of the model. Naive inference is quite slow due to the large number of keypoints and parts ($N=68+10$), and huge state space for each node which includes location, occlusion pattern and shape mixtures. Consider the message passed from one part to another where each part has L possible locations, S shape mixtures and O occlusion patterns. In general this requires minimizing over functions

1. In practice we find it is sufficient for the deformation cost to only depend on the child shape mixture, i.e. $\psi_{ij}(l_i, l_j, s_i, s_j) = w_{ij}^{s_j} \cdot \psi(l_i - l_j)$ which gives a factor S speedup with little decrease in performance.

of size $(LSO)^2$ or $L(SO)^2$ when using the distance transform. In the models we test, $SO = 12$ which poses a substantial computation and memory cost, particularly for high-resolution images where L is large.

Part-part messages: While the factorization of shape and occlusion doesn’t change the asymptotic complexity, we can reduce the runtime in practice by exploiting distributivity of the distance transform over max to share computations. Standard message passing from part j to part i requires that we compute:

$$\mu_{j \rightarrow i}(l_i, s_i, o_i) = \max_{l_j, s_j, o_j} \left[\psi_{ij}(l_i, l_j, s_i, s_j) + \sum_{k \in \text{child}(j)} \mu_{k \rightarrow j}(l_j, s_j, o_j) + b_{ij}(s_i, s_j, o_i, o_j) \right]$$

where we have dropped the unary term ϕ_j which is 0 for parts. Since the bias doesn’t depend on the location of parts we can carry out the computation in two steps:

$$\nu_{ij}(l_j, s_i, s_j, o_j) = \max_{l_j} \left[\psi_{ij}(l_i, l_j, s_i, s_j) + \sum_{k \in \text{child}(j)} \mu_{k \rightarrow j}(l_j, s_j, o_j) \right]$$

$$\mu_{j \rightarrow i}(l_i, s_i, o_i) = \max_{s_j, o_j} [\nu_{ij}(l_j, s_i, s_j, o_j) + b_{ij}(s_i, s_j, o_i, o_j)]$$

which only requires computing S^2O distance transforms.

Keypoint-part messages: In our model the occlusion and shape variables for a keypoint are determined completely by the parent part state. Since the score is known for an occluded keypoint in advance, it is not necessary to compute distance transforms for those components. We write this computation as:

$$\nu_{jk}(l_j, s_j, o_j) = \begin{cases} 0 & \text{if } k \text{ occluded in } o_j \\ \max_{l_k} \psi_{jk}(l_j, l_k, s_j, s_j) + \phi_k(l_k, s_j, o_j | I) \end{cases}$$

$$\mu_{k \rightarrow j}(l_j, s_j, o_j) = \nu_{jk}(l_k, s_j, o_j) + b_{jk}(s_j, o_j, s_j, o_j)$$

Where we have used the notation to explicitly capture the constraint that keypoint shape and occlusion mixtures (s_k, o_k) must match those of the parent part (s_j, o_j) . In our models, this reduces the memory and inference time by roughly a factor of 2, a savings that becomes increasingly significant as the number of occlusion mixtures grows.

3.3 Global Mixtures for Viewpoint and Resolution

Viewpoint and image resolution are the largest sources of variability in the appearance and relative location of keypoints. To capture this, we use a mixture over three head poses $\mathcal{V} = \{\text{left}, \text{right}, \text{center}\}$. These “global” mixtures can be represented with the same notation as above by expanding the state-space of the shape variables to be the cross product of the set of local shapes for part i and the global viewpoint for the model (i.e., $s_i \in \mathcal{S}_i \times \mathcal{V}$) and fixing some entries of the bias b_{ij} to be $-\infty$ to prevent mixing of local shapes from different viewpoints. In our

implementation we tie parameters to enforce the left- and right-facing models to be mirror symmetric.

The HPM model we have described includes a large number of keypoints. While this is appropriate for high resolution imagery, it does not perform well in detecting and modeling low resolution faces (< 150 pixels tall). To address this we introduce an additional global mixture component for each viewpoint that corresponds to low-resolution HPM model consisting of a single half-resolution template for each part and no keypoint templates. This mixture is trained jointly with the full resolution model using the strategy described in [29].

4 MODEL TRAINING AND INFERENCE

The potentials in our shape model are linearly parameterized, allowing efficient training using an SVM solver [18]. Face viewpoint, keypoint locations, shape and occlusion mixtures are completely specified by pre-clustering the training data so that parameter learning is fully supervised. We first describe how these supervised labels are derived from training data and how we synthesize “virtual” positive training examples that include additional occlusion. We then discuss the details of the parameter learning and test-time prediction.

4.1 Training Data

We assume that a training data set of face images has been annotated with keypoint locations for each face. From such data we automatically generate additional mixture labels specifying viewpoint, shape, and occlusion. We also generate additional virtual training examples by synthesizing plausible coherent occlusion patterns.

Viewpoint and Resolution Mixtures: To cluster training examples into a set of discrete viewpoints, we make use of the MultiPIE dataset [30] which provides ground-truth viewpoint annotations for a limited set of faces. We perform Procrustes alignment between each training example and examples in the MultiPIE database and then transfer the viewpoint label from nearest MultiPIE example to the training example. In our experiments we used only three viewpoint clusters: center (± 7.5 degrees), left, and right-facing (7.5 - 22.5 degrees). In addition to viewpoint, alignment to MultiPIE also provides a standard scale normalization and removes in-plane rotations from the training set. To train the low-resolution mixture components, we use the same training data but down-sampled the input image by a factor of 2.

Part Shape and Occlusion Mixtures: For each part and each viewpoint, we cluster the set of keypoint configurations in the training data in order to come up with a small number of shape mixtures for that part. The part shapes in the final model are represented by displacements relative to a parent node so we subtract off the centroid of the part keypoints from each training example prior to clustering. The vectors containing the coordinates of the centered keypoints are clustered using k-means. We

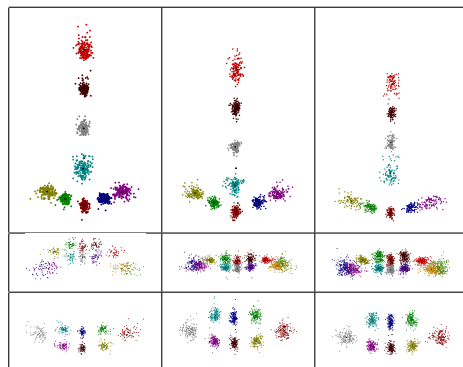


Fig. 4: Example shape clusters for face parts (nose, upper lip, lower lip). Co-occurrence biases for combinations of part shapes are learned automatically from training data. Different colored points correspond to location of each keypoint relative to the part (centroid).

imagine it would be efficient to allocate more mixtures to parts and viewpoints that show greater variation in shape, but in the final model tested here we use fixed allocation of $S = 3$ shape mixtures per part per viewpoint. Fig. 4 shows example clusterings of part shapes for the center view.

Synthetic Occlusion Patterns: In the model each keypoint is fully occluded or fully visible. The occlusion state of a part describes the occlusion of its constituent keypoints. If there are N_k keypoints then there are 2^{N_k} possible occlusion patterns. However, many of these occlusions are quite unlikely (e.g. every other keypoint occluded) since occlusion is typically generated by an occluder object with a regular, compact shape.

To model spatial coherence among the keypoint occlusions, we synthetically generate “valid” occlusions patterns by first sampling mean part and keypoint locations from the model and then randomly sampling a quarter-plane shaped occluder and setting as occluded those keypoints that fall behind the occluder. Let a, b be uniformly sampled from a tight box surrounding the face. A keypoint i with location $l_i = (x, y)$ is occluded if $(x \leq a) \wedge (y \leq b)$ where the quadrant is chosen at random. While our occluder is somewhat “boring”, it is straightforward to incorporate more interesting shapes, e.g., by sampling from a database of segmented objects. Fig. 3 shows example occlusions generated for a training example.

In our experiments we generate 8 synthetically occluded examples for each original training example. For each part in the model we cluster the set of resulting binary vectors in order to generate a list of valid part occlusion patterns. The occlusion state for each keypoint in a training example is then set to be consistent with the assigned part occlusion pattern. In our experiments we utilized only $O = 4$ occlusion mixtures per part, typically corresponding to unoccluded, fully occluded and two half occluded states whose structure depended on the part shape and location within the face.



Fig. 5: Examples of landmark localization and occlusion estimation for images from the HELEN (row 1) and COFW (rows 2-3) test datasets. Red indicates those keypoints which are predicted as being occluded by the HPM.

4.2 Parameter learning

Writing our scoring function on training image I as an inner product of weights and features $Q(l, s, o) = w \cdot \Psi(l, s, o|I)$, we solve the regularized SVM

$$\min_w \frac{1}{2} \|w\|^2 + C \sum_t \eta_t$$

$$w \cdot \Psi(l^t, s^t, o^t|I^t) \geq 1 - \eta_t \quad \forall t \in \mathcal{P}$$

$$w \cdot \Psi(l, s, o|I^t) \leq -(1 - m\delta(o) - \eta_t) \quad \forall l, s, o \quad \forall t \notin \mathcal{P}$$

where (l^t, s^t, o^t) denotes the supervised model configuration for a positive training example, $\delta(o)$ is a margin scaling function that measures the fraction of occluded keypoints and C and m are hyper-parameters. The constraint on positive images $t \in \mathcal{P}$ encourages that the score of the correct model configuration be larger than 1 and penalizes violations using slack variable η_t . The second constraint encourages the score to be low on all negative training images $t \notin \mathcal{P}$ for all configurations of the latent variables.

Margin scaling for occlusion: This formulation differs from standard supervised DPM training in the treatment of negative training examples. Since keypoints can be occluded in our model, fully or partially occluded faces can be detected by our model in the negative images. These images do not contain any faces and we would like our model generates low scores for these detections. However, a keypoint which is detected as occluded in a negative image is in some sense correct. There is no real distinction between a negative image and a positive image of a fully occluded face! Thus we penalize negative detections (false positives) with significant amounts of occlusion less than fully visible false positives.

For this purpose, we scale the margin for negative examples in proportion to the number of occluded key-

points. We specify the margin for a negative example as $1 - m\delta(o)$, where the function $\delta(o)$ measures the fraction of occluded keypoints and m is a parameter. As the number of occluded keypoints increases the margin decreases and the model score for that example can be larger without violating the constraint. The margin for a fully occluded example is equal to $1 - m$. Setting $m = 0$ corresponds to standard classification where all the negatives have the same margin of 1. In this case the biases learned for occluded keypoints tend to be low, otherwise many fully or partially occluded negative examples will violate the constraint. As a result, models trained with $m = 0$ tend not to predict occlusion. As we increase m , the scores of fully or partially occluded negative examples can be larger without violating the constraint and the training procedure is thus free to learn larger bias parameters associated with occluded keypoints. As we show in our experimental evaluation, this results in higher recall of occluded keypoints and improved test-time performance.

We use a standard hard-negative mining or cutting-plane approach to find a small set of active constraints for each negative image. Given a current estimate of the model parameters w , we find the model configuration (s, l, o) that maximizes $w \cdot \Psi(l, s, o|I^t) - m\delta(o)$ on a negative window I^t . Since the loss $m\delta(o)$ can be decomposed over individual keypoints, this loss-augmented inference can be easily performed using the same inference procedure introduced in section 3. We simply subtract $\frac{m}{N_k}$ from the messages sent by occluded keypoints where $N_k = 68$ is the number of keypoints. During training we make multiple passes through the negative training set and maintain a pool of hard negatives for each image. We share the slack variable η_t for all such negatives found over a single window I_t .

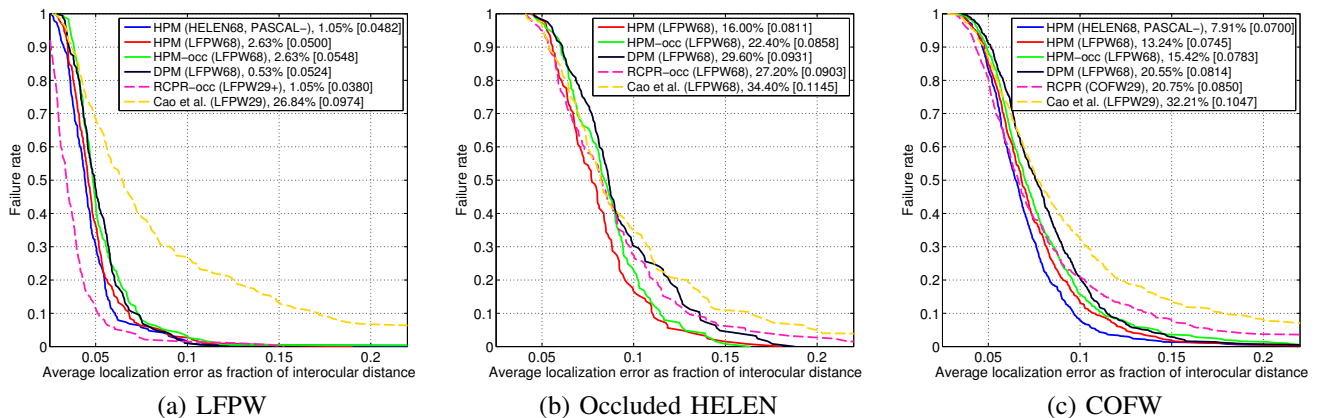


Fig. 6: Panels show cumulative error distribution curves (the proportion of test images that have average landmark localization error below a given threshold) on three test sets: LFPW, an occlusion rich subset of HELEN and on COFW. We compare the hierarchical part model (HPM) with and without occlusion mixtures (HPM-occ) to a baseline tree-structured DPM [3] with the same set of keypoints, and to robust pose regression (RCPR) [15]. The legend indicates the training set (in parentheses), the failure rate % at a localization threshold of 0.1 and the average error [in brackets]. The HPM shows good localization performance, especially on more difficult datasets with significant occlusion. In general regression models (dashed lines) have better performance for a low localization threshold compare to part based models (solid lines). However, the failure rates for regression models drop more slowly and eventually cross over those for part models (solid lines) as the allowable localization error threshold increases.

4.3 Test-time Prediction

Scale and In-plane Rotation: In our experiments, we observed that part models with standard quadratic spring costs are surprisingly sensitive to in-plane rotation. Models that performed well on images with controlled acquisition (such as MultiPIE) fared poorly “in the wild” when faces were tilted. The alignment procedure described above explicitly removes scale and in-plane rotations from the set of training examples. At test time detection, we perform an explicit search over scale and in plane rotations (-30 to 30 degrees with an increment of 6 degrees).

Landmark Prediction: To benchmark keypoint localization of the model across datasets that utilized different landmark annotation standards, we used linear regression to learn a mapping from the set of locations returned by our hierarchical part model. In our experiments, this regression was important to accurately benchmark localization performance. Using a heuristic approach of simply taking the closest keypoint reported performed significantly worse, in some cases doubling failure rates.

Let $l^i \in \mathbb{R}^{2N_k}$ be the vector of keypoint locations returned when running the model on a training example i and $\hat{l}^i \in \mathbb{R}^{2M}$ a vector of ground-truth keypoint location for that image. We train a linear regressor

$$\min_{\beta} \sum_i \|\hat{l}^i - \beta^T l^i\|^2 + \lambda \|\beta\|^2$$

where $\beta \in \mathbb{R}^{2N_k \times 2M}$ is the matrix of learned coefficients and λ is a regularization parameter. To prevent overfitting, we restrict β_{pq} to be zero unless the keypoint p belongs to the same part as q .

To predict keypoint occlusion, we carried out a similar mapping procedure using regularized logistic regression.

However, we found that in practice a much simpler rule of specifying a correspondence between the two sets of keypoints based on their distance and transferring the occlusion flag from the model to benchmark keypoints achieved the same accuracy.

5 EXPERIMENTAL EVALUATION

Figure 5 shows example outputs of the HPM model run on example face images. The model produces both a detection score and estimates of keypoint locations and occlusion states. While the possible occlusion patterns are quite limited (4 occlusions patterns per part shape), the final predicted occlusions (marked in red) are quite satisfying in highlighting the support of the occluder for many instances. We evaluate the performance of the model on three different tasks: landmark localization, landmark occlusion prediction, and face detection. In our experiments we focus on test datasets that have significant amounts of occlusion and emphasize the ability of the model to generalize well across datasets.

5.1 Landmark Localization

We evaluate performance of our method and related baselines on three benchmark datasets for keypoint localization: Labeled Face Parts in the Wild (LFPW) [31], a subset of the HELEN dataset [32] which contained occlusions, and the more difficult Caltech Occluded Faces in the Wild (COFW) [15] dataset. The latter two datasets were selected to highlight the performance of our model in the presence of occlusion and a wider variety of poses. The authors of [15] estimate that LFPW only contains 2% occluded keypoints compared to 23% for COFW. Fig. 5 depicts

model	training dataset	LFPW (29)		COFW (29)	
		FR	AE	FR	AE
RCPR-occ	LFPW29	11.05	0.073	36.56	0.115
RCPR-occ	LFPW29+	1.05	0.038	36.36	0.096
RCPR-occ	COFW29	10.99	0.071	23.72	0.091
RCPR	COFW29	8.95	0.064	20.75	0.085
HPM	LFPW68,INR-	2.63	0.050	13.24	0.075
HPM	HELEN68,INR-	1.58	0.049	9.29	0.072
HPM	HELEN68,PAS-	1.05	0.048	7.91	0.070

TABLE 1: We find HPM generalizes well across datasets while pose regression has a strong dependence on training data. Localization performance is measured by failure rate (FR) and average error (AE). The RCPR model trained on COFW performs much better on COFW test data compared to RCPR-occ trained on LFPW29+ (20% FR vs 36% FR) but has much worse performance on LFPW test data compared to that model (8% FR vs 1% FR). Good performance on LFPW also depends heavily on including additional warped positive instances (LFPW29+ vs LFPW29). The HPM trained on LFPW68 has low failure rates on both COFW (13%) and LFPW (2%) test data. Last two rows of the table show the performance of HPM when a different training data set (HELEN68) is used for training. This dataset has more variation and more images (1758) compared to LFPW68 (682) and improves performance of HPM on both test datasets. Training on more negative images (6000 images from PASCAL) decreases localization error of our model compared to using only INRIA negatives.

selected results of running our detector on example images from the HELEN and COFW test datasets.

We note there is a variety of keypoint annotation conventions across these different datasets. LFPW and COFW contain a set of 29 landmarks while HELEN includes a much denser set of 194 landmarks. The 300 Faces in-the-wild Challenge (300-W) [33] has also produced several unified benchmarks in which the LFPW and HELEN datasets have been re-annotated with a set of 68 standard keypoints. For the purposes of benchmarking, and to allow easy comparison to previously reported work, we utilize the 29 keypoints for the LFPW and COFW datasets and the 68 keypoints for HELEN.

Localization Evaluation Metrics: To evaluate keypoint localization independent of detection accuracy, we assume that detection has already been performed and run the algorithm on cropped versions of the test images. While our model is capable of both detecting and localizing keypoints, we would like to compare to pose regression methods that require good initialization. We thus follow the standard protocol (see e.g., [33]) of using the ground-truth face bounding box to initialize the system. For pose regression methods we randomly generate an initial bounding box that overlaps the ground-truth face box by at least 80%. For our model, we evaluate the localization accuracy for the highest scoring detection that overlaps the ground-truth by at least 80%.

We report the average keypoint localization error across

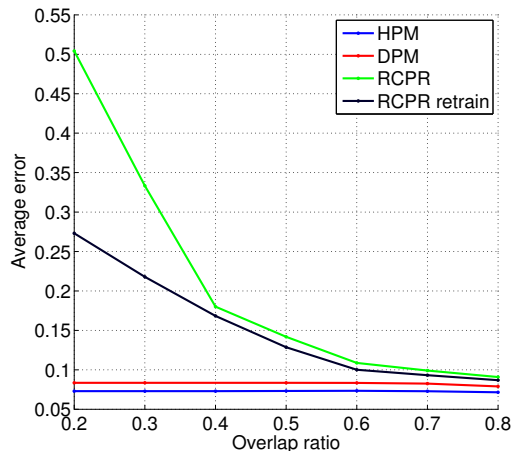


Fig. 7: We analyze the landmark localization average error of RCPR, HPM and DPM for different overlap ratio with the ground-truth face boxes. For RCPR we change the minimum overlap ratio of the initial bounding boxes and the ground-truth face boxes. For HPM and DPM, we change the minimum overlap threshold of the returned detections and ground-truth boxes. RCPR is very sensitive to the amount of overlap and its performance decreases rapidly as the overlap ratio decreases. But, HPM and DPM are robust to the overlap threshold and they can maintain the same performance over different thresholds.

the entire test set as well as the proportion of “failures”, test images that had average keypoint localization above a given threshold. Distances used in both quantities are expressed as a proportion of the inter-ocular (iod) distance specified by the ground-truth. Computing the failure rate across a range thresholds yields a cumulative error distribution curve (Fig. 6). When a single summary number is required we report the failure rate at a standard threshold of 0.1 iod.

We note that there is some basic difficulty in benchmarking localization on heavily occluded examples. The COFW test data includes human-marked annotations for occluded keypoints but exact placement of occluded keypoints is difficult even for humans. It may be more informative for applications such as face identification to only benchmark localization of visible keypoints. In the following experiments we use the standard benchmark protocol that includes occluded keypoints but note that the localization failure rates drop by half (e.g., on COFW) when only benchmarking localization accuracy of visible keypoints.

Training and baselines: To train our model, we used a set of 682 near-frontal training images taken from LFPW using the 68 keypoint annotations provided by 300-W. From each training image we generate 8 synthetically occluded “virtual positives” yielding a final training set of 6138 positives. As mentioned previously, since we explicitly search over in-plane rotations and scales, standardize the pose of each training image prior to learning the model. To evaluate cross-dataset generalization, we also trained a version of our model on a portion the HELEN dataset consisting of 1758 frontal images annotated with 68 keypoints. To fit

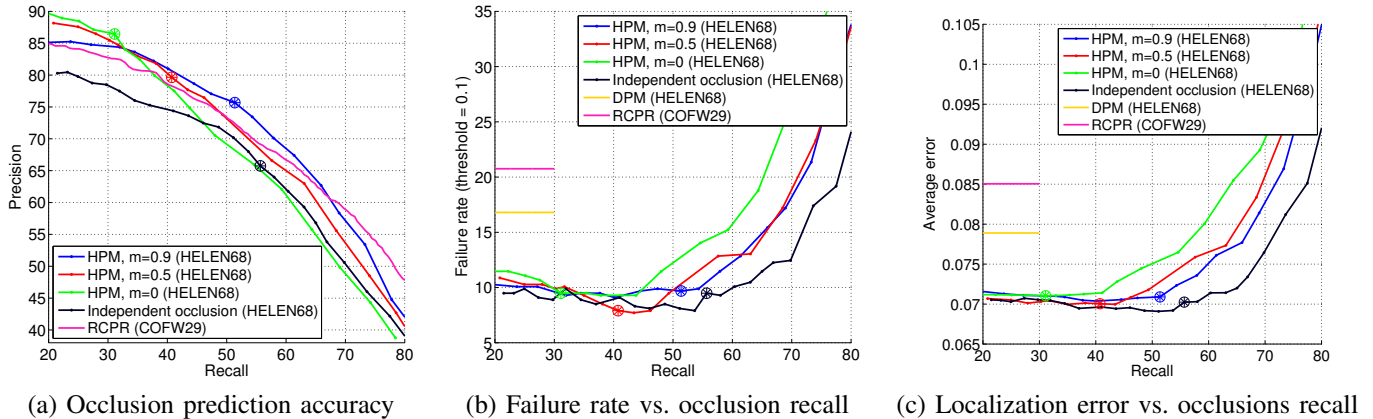


Fig. 8: Occlusion prediction accuracy on the COFW test dataset for variants of our model. Using a suitable margin scaling function (see Sec. 4.2) allows for significantly better occlusion prediction accuracy (a) over an independent occlusion model (a) with minimal loss in localization performance (b,c). Localization performance of DPM and RCPR are included for reference.

linear regression coefficients for mapping from the HPM predicted keypoint locations to 29 keypoint datasets, we ran the model on the COFW training data set and fit regression parameters β that mapped from the 68 predicted points to the 29 annotated.

For comparison, we trained baseline models including a version of our model without occlusion mixtures (HPM-occ) and the non-hierarchical deformable part model² (DPM) described by [3]. We also evaluate the robust cascaded pose regression (RCPR) described in [15] and their implementation of explicit shape regression [16] using both pre-trained models provided by the authors and models retrained to predict 68 keypoints. Unlike HPM which uses virtual occlusion, RCPR requires training examples with actual occlusions and corresponding annotations. For training sets that featured no occlusion, we thus trained a variant that does not model occlusion (RCPR-occ).

Localization Results (LFPW): Labeled Face Parts in the Wild (LFPW) [31], a commonly used dataset for evaluating landmark estimation consisting of 300 test images annotated with a standard set of 29 keypoints. The original LFPW test set is no longer completely available due to broken links, but we were able to download 194 of the test images. Fig. 6(a) shows the localization error distribution.

For the LFPW dataset, which features relatively little occlusion, different part-based models perform similarly while pose regression yielded better results at tight localization thresholds. The HPM model achieved an average localization error of 0.05 when trained on LFPW and slightly lower (0.048) when trained on HELEN. By comparison, the DPM model [3] has a higher average error 0.052. The robust pose regression model of [15] yielded a very low average error of 0.038. However, we have observed that the exact choice of training data is essential to achieving this

level of performance. As highlighted in Table 1 training on other datasets resulted in significantly worse performance while HPM performance was quite stable across datasets. In addition, achieving good performance with RCPR on LFPW also required boosting the set of training images by adding 2000 perturbed versions of training images as additional training examples (denoted LFPW29+ in the table).

Localization Results (HELEN): We evaluated on a subset of the HELEN dataset [32] consisting of 126 images which were selected on the basis having some significant amount of occlusion. HELEN generally includes more difficult images than LFPW and our selected subset was harder still. These test images contain 68 keypoint annotations so we evaluated only models trained on LFPW68. We do not report results of the HPM (HELEN68) on this dataset as there was overlap between training and testing images. Fig. 6(b) shows the error distribution. The HPM achieves an average error of 0.0811, beating out the DPM baseline (0.0931) and RCPR (0.0903). Removing explicit occlusion from the model (HPM-occ) results in higher failure rates for a range of thresholds.

Localization Results (COFW): Finally, we tested on the 507 image test set from Caltech Occluded Faces in the Wild (COFW) [15] containing internet photos depicting a wide variety of more difficult poses and includes a significant amount of occlusion. Since COFW training only contains 29 keypoints, we could not train the HPM model and instead evaluate models trained on LFPW68 and HELEN68. Fig. 6(c) shows that HPM achieves a significantly lower average error than RCPR and lower failure rates for all but the smallest (< 0.06) localization failure thresholds.

Dependence of Localization on Detection: A key benefit of the HPM (and DPM [3]) approach is that the same model serves to both detect and localize the keypoints. In contrast, pose regression methods such as RCPR require that the face already be detected. This distinction becomes particularly

2. The original DPM model of [3] was trained on the very constrained MultiPIE dataset [30]. Retraining the model and performing a similar search over in-plane rotations yielded significantly better performance which we report here (c.f., [15])

important for occluded faces since detection is significantly less accurate (see Detection experiments below).

To characterize the dependence of keypoint localization on accurate detection, we benchmarked average localization error for varying degrees of overlap between the hypothesized detection and ground-truth bounding box on the COFW test set. As shown in Fig. 7, decreasing the overlap ratio has no affect HPM / DPM performance since there are never false positives in the vicinity of the face that score higher than one with high overlap ratio. In contrast, RCPR performs significantly worse when initialized from bounding boxes that do not have high overlap with the face. Since the area over which RCPR searches is learned from training data, we also retrained a version of RCPR for each degrees of overlap. This yielded improved performance but still shows a significant fall off in performance compared to the HPM.

5.2 Occlusion Prediction

To evaluate the ability of the model to correctly determine which keypoints are occluded, we evaluate the accuracy of occlusion as a binary prediction task. For a given test set, we compute precision and recall of occlusion predictions relative to the ground-truth occlusion labels of the keypoints.

We trace out a precision-recall curve for occlusion prediction by adjusting the model parameters to induce different predicted occlusions. As described in section 3, the bias parameter $b_{ij}(s_i, s_j, o_i, o_j)$ favors particular co-occurrences of part types. By increasing (decreasing) the bias for occluded configurations we can encourage (discourage) the model to use those configurations on test. Let $b_{ij}(s_i, s_j, o_i, o_j)$ be a learned bias parameter between an occluded leaf and its parent. To make the model favor occluded parts, we modify this parameter to $b_{ij}(s_i, s_j, o_i, o_j) + \text{abs}(b_{ij}(s_i, s_j, o_i, o_j)) \times \alpha$.

Fig. 8(a) depicts occlusion precision-recall curves generated by running the HPM model for different bias α offsets. The crosses mark the precision-recall for the default operating point when $\alpha = 0$. We compare performance of the HPM model with different values of the margin scaling hyper-parameter m as well as RCPR and a baseline independent occlusion model. Fig. 8 (b) and (c) show the corresponding average errors and failure rates for these models parameterized by the recall of occlusion. For large values of α , the model predicts more occlusions, resulting in improved recall at the expense of precision (a) and ultimately lower localization accuracy (b,c).

Margin scaling: As described in section 4.2, we can change the learning parameter m to produce models with different recall of occlusions at the trained operating point ($\alpha = 0$). When $m = 0$ all the negative examples including fully or partially occluded configurations are penalized equally. Therefore, model learns small biases for occluded configurations, reducing the total loss over occluded negative examples and decreasing default recall of occlusion. When driven to predict more occlusion by increasing α the

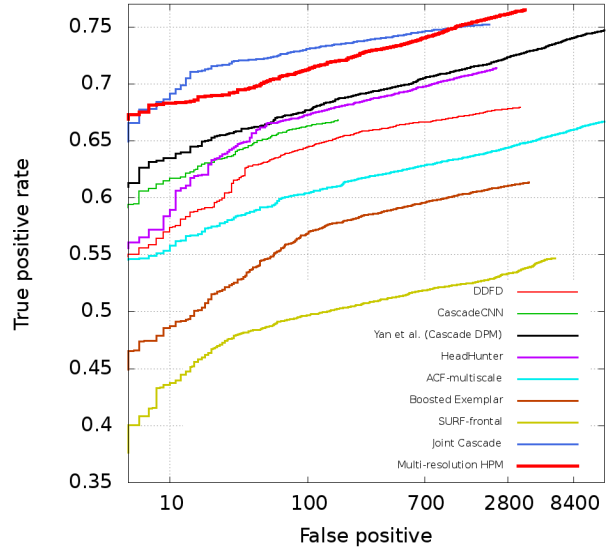


Fig. 9: Face detection performance on the continuous-ROC FDDDB benchmark [5]. Multi-resolution HPM is trained on the front view training data, while this dataset has many side view faces. But still it has a competitive performance when compared with state-of-the-art face detection methods trained on front view and side view training data [34].

model localization performance degrades rapidly. Training the model with larger values of m yields a model which naturally predicts occlusion more frequently and degrades more gracefully for larger values of α . We found that choosing a value of $m = 0.5$ provided a good compromise, improving both recall and localization accuracy.

Independent occlusion baseline: We compared the results of HPM with a model that had the same architecture but in which there are no occlusion mixtures at the part level and each keypoint is allowed to be independently set to visible or occluded depending on learned biases. We refer to this as “independent occlusion” since the model does not capture any correlations between the occlusion of different keypoints. We found that this independent occlusion model has many of the same benefits as the HPM model in terms of keypoint localization accuracy (Fig. 8). However, occlusion prediction accuracy is significantly worse in the independent model with precisions typically 5% lower than HPM($m = 0.9$) over a range of recall values.

5.3 Detection

Pose regression requires good initialization provided by a face detector to accurately locate keypoints. In contrast, part-based models have the elegant advantage of performing detection and localization simultaneously. In this section, we compare the detection performance of our approach and other top methods on two datasets: FDDDB [5] and our own Occluded Face Detection (OFD-UCI) dataset.

Multi-resolution HPM: Since many face detection datasets such as FDDDB contain many low-resolution faces, we trained a multi-resolution variant of our model [29]. This

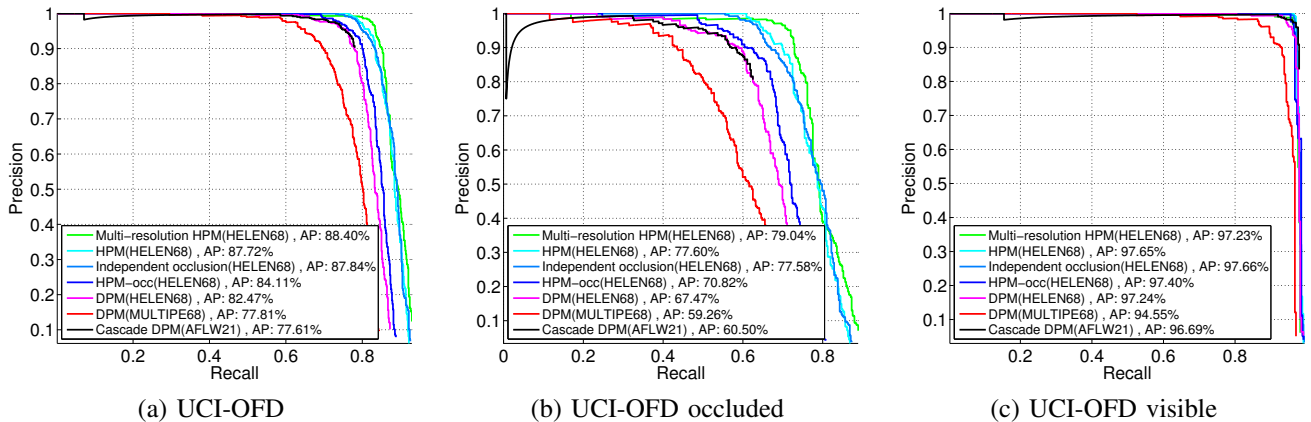


Fig. 10: Precision-Recall curves of face detection on our UCI-OFD dataset (a) for all of the faces, (b) occluded subset and (c) visible subset. On the visible subset our model, DPM retrained on HELEN68 and Cascade DPM [35] have almost similar performances, but our model significantly outperforms these methods on the occluded subset and it has a better overall performance. Cascade DPM uses many accelerate techniques, which may reject some of the faces. Its maximum recall for the visible faces is near 100%, while its maximum recall for the occluded faces is only 60%. The initial drop in the Precision-Recall of this method for the occluded subset is because its returned bounding boxes for some of the high scored occluded faces are not accurate and do not have the minimum 0.5 overlap with the ground-truth bounding boxes.

model has a high and a low-resolution model for each viewpoint. The high resolution model has the same structure as our trained model for landmark localization except that parts are represented as 3x3 HoG cells rather than 5x5. The low-resolution model has 7 parts (right eye, left eye, nose, mouth, chin, left jaw and right jaw) each of which is represented by 7x7 HoG cells with the spatial bin size of 4. Each part has one shape mixture and 2 occlusion mixtures (visible or occluded). The heights (eyebrow to chin) of the large model and small model are about 100 and 60 pixels respectively. To detect even smaller images, we upsample input images by a factor of 2 to allow for detection of faces as small as 30 pixels. We trained this model using the same 1758 positive examples from HELEN68 and generated 8 virtual positive examples per example. For negative images we used 6000 images from the PASCAL VOC 2010 trainval set which do not contain people.

Detection on Fddb: We evaluated our multi-resolution model on the widely used Fddb dataset. This dataset contains 5171 faces in a set of 2845 images. Faces are annotated by ellipses in this dataset and are as small as 20 pixels in height. To match that, we map our predicted keypoint locations to ellipses using a linear regression model. Fddb has 10 folds and the ROC curves are the average over the results of these folds. To compute ellipses for each fold, we learnt the linear regression coefficients using examples from the other 9 folds.

We used the standard evaluation protocol for this dataset and compared our method with the top published results available on the Fddb website [34]. The continuous ROC curve for our method and leading methods are shown in Fig. 9 plotted on a semi-log scale. Our result is highly competitive with the top results. The model has better performance on the continuous ROC evaluation relative to

other methods because it can predict location of parts and compute accurate bounding ellipses around the faces.

We point out that our model is trained on near frontal faces and only includes mixtures for near frontal viewpoints, while this dataset has many side view faces. Because the HPM can handle occlusion it still can detect some side view faces as long as the appearance of the visible parts is close to their frontal view appearance. We believe adding more viewpoint to the model will likely improve performance further on this dataset.

UCI Occluded Face Detection (OFD-UCI) Dataset:

Fddb has a few occluded faces, but most of the faces are fully visible with many low resolution and side view faces. In order to better measure the ability of our model to handle detection of occluded faces, we assembled a preliminary dataset for occluded face detection consisting of 61 images from Flickr containing 766 labeled faces. Of the faces in these images, 430 include some amount of occlusion. Most of the faces are near frontal and vertical. Height (eyebrow to chin) of the smallest face is about 40 pixels. Precision/Recall curves of face detection of multi-resolution HPM, HPM, HPM-occ, DPM and Cascade DPM [35] are shown in Fig. 10(a). We further break down performance, plotting Precision/Recall curves for the subset of faces with some amount of occlusion in (b) and fully visible in (c). Precision and recall for occluded subset of faces are calculated as below:

$$\text{Precision}_o = \frac{tp_o}{tp_o + fp}, \text{Recall}_o = \frac{tp_o}{tp_o + fn_o}$$

where tp_o and fn_o show number of correct detection and miss detection of occluded faces, respectively. Our method significantly outperforms other methods on the occluded subset and the performance of all of the methods are almost equal on the visible subset. Fig. 11 shows example

detection results produced by the model on cluttered scenes containing many overlapping faces.

6 DISCUSSION AND CONCLUSION

Our experimental results demonstrate that adding coherent occlusion and hierarchical structure allows for substantial gains in performance for keypoint localization and detection in part models. Our final HPM outperforms previously published results on the challenging COFW dataset in terms of keypoint localization accuracy and shows robust generalization across different training and test sets.

In images with relatively little occlusion, the HPM gives similar detection and localization performance to other part-based approaches, e.g. DPM, but is significantly more robust to occlusion. Our results also suggest that when it is useful to determine exactly which parts are occluded (e.g., for later use in face identification), independent occlusion makes weaker predictions than our part occlusion mixtures which enforce coherence between neighboring keypoints.

In comparing pose regression and part-based models, there seem to be several interesting trade-offs. In our experiments, we see a general trend in which error distribution curves for pose regression and part-based models cross, suggesting that RCPR yields very accurate localization for a subset of images relative to the HPM but fails for some other proportion even at very large error thresholds. Unlike pose regression, the part model performs detection, eliminating the need for detection as a pre-process and improving robustness. In particular, we are able to detect many heavily occluded faces which would not be detected by a standard cascade detector. We find that the HPM tends to generalize well across datasets suggesting it avoids some overfitting problems present in pose regression.

The run-time of our model implementation built on dynamic programming lags significantly behind those of regression-based, feed-forward approaches. Our model takes ~ 10 s to run on a typical COFW image, roughly 100x slower than RCPR. On the other hand, pose regression depends critically on having good initialization while the part model approach can be used for both simultaneous detection and localization.

Finally, we note that there are many avenues for future work. Performance depends on the graphical independence structure of the model which should ideally be learned from data. While our model implicitly represents the pattern of part occlusions, it does not integrate local image evidence for the occluder itself. A natural extension would be to add local filters which detect the presence of an occluding contour between the occluded and non-occluded keypoints. Such filters could be shared across parts to avoid increasing too much the overall computation cost while moving closer to our goal of explaining away missing object parts using positive evidence of coherent occlusion.

REFERENCES

- [1] V. Blanz and T. Vetter, "Face recognition based on fitting a 3d morphable model," *PAMI*, vol. 25, no. 9, pp. 1063–1074, 2003. 1, 2
- [2] A. Martinez and S. Du, "A model of the perception of facial expressions of emotion by humans: Research overview and perspectives," *JMLR*, vol. 98888, pp. 1589–1608, 2012. 1
- [3] X. Zhu and D. Ramanan, "Face detection, pose estimation, and landmark localization in the wild," in *CVPR*, 2012, pp. 2879–2886. 1, 2, 3, 7, 9
- [4] G. Ghiasi and C. Fowlkes, "Occlusion coherence: Localizing occluded faces with a hierarchical deformable part model," in *CVPR*, 2014, pp. 1899–1906. 2
- [5] V. Jain and E. Learned-Miller, "Fddb: A benchmark for face detection in unconstrained settings," University of Massachusetts, Amherst, Tech. Rep. UM-CS-2010-009, 2010. 2, 10
- [6] A. L. Yuille, P. W. Hallinan, and D. S. Cohen, "Feature extraction from faces using deformable templates," *IJCV*, vol. 8, no. 2, pp. 99–111, 1992. 2
- [7] T. F. Cootes, G. J. Edwards, and C. J. Taylor, "Active appearance models," *PAMI*, vol. 23, no. 6, pp. 681–685, 2001. 2
- [8] I. Matthews and S. Baker, "Active appearance models revisited," *IJCV*, vol. 60, no. 2, pp. 135–164, 2004. 2
- [9] S. Milborrow and F. Nicolls, "Locating facial features with an extended active shape model," in *ECCV*, 2008, pp. 504–513. 2
- [10] L. Wiskott, J.-M. Fellous, N. Kuiger, and C. Von Der Malsburg, "Face recognition by elastic bunch graph matching," *PAMI*, vol. 19, no. 7, pp. 775–779, 1997. 2
- [11] L. Gu and T. Kanade, "3d alignment of face in a single image," in *CVPR*, 2006, pp. 1305–1312. 2
- [12] P. N. Bellhumeur, D. W. Jacobs, D. J. Kriegman, and N. Kumar, "Localizing parts of faces using a consensus of exemplars," in *CVPR*, 2011, pp. 545–552. 2
- [13] M. Valstar, B. Martinez, X. Binefa, and M. Pantic, "Facial point detection using boosted regression and graph models," in *CVPR*, 2010, pp. 2729–2736. 2
- [14] B. Efraty, C. Huang, S. K. Shah, and I. A. Kakadiaris, "Facial landmark detection in uncontrolled conditions," in *IJCB*, 2011, pp. 1–8. 2
- [15] X. P. Burgos-Artizzu, P. Perona, and P. Dollár, "Robust face landmark estimation under occlusion." *ICCV*, 2013. 2, 3, 7, 9
- [16] X. Cao, Y. Wei, F. Wen, and J. Sun, "Face alignment by explicit shape regression," in *CVPR*, 2012, pp. 2887–2894. 2, 9
- [17] M. Dantone, J. Gall, G. Fanelli, and L. Van Gool, "Real-time facial feature detection using conditional regression forests," in *CVPR*, 2012, pp. 2578–2585. 2
- [18] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *PAMI*, vol. 32, no. 9, pp. 1627–1645, 2010. 2, 3, 5
- [19] Y. Yang and D. Ramanan, "Articulated human detection with flexible mixtures-of-parts," *IEEE TPAMI*, 2013. 2, 4
- [20] L. Zhu, Y. Chen, and A. Yuille, "Learning a hierarchical deformable template for rapid deformable object parsing," *PAMI*, vol. 32, no. 6, pp. 1029–1043, 2010. 2
- [21] R. B. Girshick, P. F. Felzenszwalb, and D. A. Mcallester, "Object detection with grammar models," in *NIPS*, 2011, pp. 442–450. 2
- [22] M. Weber, M. Welling, and P. Perona, "Towards automatic discovery of object categories," in *Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on*, vol. 2. IEEE, 2000, pp. 101–108. 2
- [23] R. Fergus, P. Perona, and A. Zisserman, "Object class recognition by unsupervised scale-invariant learning," in *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, vol. 2. IEEE, 2003, pp. II–264. 2
- [24] H. Azizpour and I. Laptev, "Object detection using strongly-supervised deformable part models," in *ECCV*, 2012, pp. 836–849. 2
- [25] X. Chen and A. Yuille, "Parsing occluded people by flexible compositions," in *Computer Vision and Pattern Recognition (CVPR)*, 2015. 3
- [26] J. M. Saragih, S. Lucey, and J. F. Cohn, "Deformable model fitting by regularized landmark mean-shift," *IJCV*, vol. 91, no. 2, pp. 200–215, 2011. 3
- [27] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *CVPR*, 2005, pp. 886–893. 3
- [28] G. Ghiasi, Y. Yang, D. Ramanan, and C. Fowlkes, "Parsing occluded people," in *CVPR*, 2014, pp. 2401–2408. 4
- [29] D. Park, D. Ramanan, and C. Fowlkes, "Multiresolution models for object detection," in *ECCV*, 2010, pp. 241–254. 5, 10
- [30] R. Gross, I. Matthews, J. Cohn, T. Kanade, and S. Baker, "Multi-PIE," *IVC*, vol. 28, no. 5, pp. 807–813, 2010. 5, 9

- [31] M. Kostinger, P. Wohlhart, P. M. Roth, and H. Bischof, “Annotated facial landmarks in the wild: A large-scale, real-world database for facial landmark localization,” in *ICCV Workshop*, 2011, pp. 2144–2151. [7](#), [9](#)
- [32] V. Le, J. Brandt, Z. Lin, L. Bourdev, and T. S. Huang, “Interactive facial feature localization,” in *ECCV*, 2012, pp. 679–692. [7](#), [9](#)
- [33] M. Pantic, G. Tzimiropoulos, and S. Zafeiriou, “300 faces in-the-wild challenge (300-w),” in *ICCV Workshop*, 2013. [8](#)
- [34] <http://vis-www.cs.umass.edu/fddb/results.html>. [10](#), [11](#)
- [35] J. Yan, Z. Lei, L. Wen, and S. Z. Li, “The fastest deformable part model for object detection,” in *CVPR*, 2014, pp. 2497–2504. [11](#)



Golnaz Ghiasi received a BS in Computer Engineering in 2007 and a MS in Artificial Intelligence in 2010 from Amirkabir University of Technology. She is currently a PhD student in Computer Science at the University of California, Irvine. Her research interests are in computer vision and machine learning.



Charless C. Fowlkes received a BS with honors from Caltech in 2000 and a PhD in Computer Science from the University of California, Berkeley in 2005, where his research was supported by a US National Science Foundation Graduate Research Fellowship. He is currently an Associate Professor in the Department of Computer Science at the University of California, Irvine. His research interests are in computer and human vision, and in applications to biological image analysis. He received the US National Science Foundation CAREER award in 2013 and the Marr Prize in 2009.



Fig. 11: Examples of detection and localization for images from our UCI-OFD dataset (rows 1-2) and images containing occlusion from FDDB dataset (rows 3-4). Detections indicated with only 7 keypoints correspond to responses from the low-resolution model component. Ellipses are predicted on FDDB images by linear regression from keypoint locations to ellipse parameters.