Parsing Human Motion with Stretchable Models

Benjamin Sapp    David Weiss    Ben Taskar
University of Pennsylvania
Philadelphia, PA 19104
{bensapp,djweiss,taskar}@cis.upenn.edu

Abstract

We address the problem of articulated human pose estimation in videos using an ensemble of tractable models with rich appearance, shape, contour and motion cues. In previous articulated pose estimation work on unconstrained videos, using temporal coupling of limb positions has made little to no difference in performance over parsing frames individually [8, 28]. One crucial reason for this is that joint parsing of multiple articulated parts over time involves intractable inference and learning problems, and previous work has resorted to approximate inference and simplified models. We overcome these computational and modeling limitations using an ensemble of tractable submodels which couple locations of body joints within and across frames using expressive cues. Each submodel is responsible for tracking a single joint through time (e.g., left elbow) and also models the spatial arrangement of all joints in a single frame. Because of the tree structure of each submodel, we can perform efficient exact inference and use rich temporal features that depend on image appearance, e.g., color tracking and optical flow contours. We propose and experimentally investigate a hierarchy of submodel combination methods, and we find that a highly efficient max-marginal combination method outperforms much slower (by orders of magnitude) approximate inference using dual decomposition. We apply our pose model on a new video dataset of highly varied and articulated poses from TV shows. We show significant quantitative and qualitative improvements over state-of-the-art single-frame pose estimation approaches.

1. Introduction

We focus on the task of estimating and tracking articulated 2D human pose in videos “in the wild”: single-view, uncontrolled settings typical in movies, television and amateur video. Reliable parsing of human motion in such videos could vastly improve and refine action recognition and semantic retrieval. This task is made difficult by the considerable background clutter, camera movement, motion blur, poor contrast, body pose and shape variation, as well as illumination, clothing and appearance diversity. There has been an explosion of recent work on articulated pose estimation from single images of this type [20, 15, 7, 16, 8, 6, 1, 21]. Despite steady advances, localization of the most interesting parts (lower arms and hands) remains very inaccurate. Video provides additional motion cues, yet most work on articulated tracking requires manual initialization from a few frames [10, 4, 2, 24, 18, 5]. Several recent papers [23, 12, 17, 8, 28], however, combine tracking and estimation without any supervised initialization; our work follows this setting, which we call articulated motion parsing.

In the “strike-a-pose" parsing method of [17], an easily detectable canonical pose (e.g., limbs spread out) is automatically found in a long stretch of video and used as initialization for person-specific part appearance models. While this intuitive idea works well when every person strikes the easy canonical pose at least once in every video, many motion sequences are short and all the poses are difficult. Most work to date on short, difficult motions does not show significant improvement over single frame parsing [8, 28], and sometimes causes actual degradation in accuracy when pose estimation is coupled across frames. One crucial reason for this is that joint parsing of multi-
ple articulated parts over time involves intractable inference and learning problems, since part location/orientation variables have very large state spaces and the models are highly inter-connected (high-treewidth). Because of this barrier, previous work has resorted to approximate inference using sampling [27, 9, 24, 23] and variational [22, 8] methods, which often introduce poorly understood error and/or bias. The computational complexity of learning such models often limits the ability to learn rich features, resulting in using only simple, image-independent location-persistence coupling [8]. One notable contrast to this is the work of [18], which uses powerful image cues and inference to establish correspondences between frames and within frames. However, their method makes greedy decisions as it proceeds through frames in order to handle the intractability of maintaining a distribution of beliefs throughout the video sequence, and hence has no way of recovering from bad choices in earlier frames.

Computational considerations also usually lead to restrictive simplifying assumptions on geometry: 2D limb lengths are fixed (given global scale) [17, 8, 6, 1]. In typical video sequences this assumption is almost always violated because of foreshortening and body type variation, especially for the lower arms. For this paper, we collected a new challenging video dataset which extends the one in [28], which we call VideoPose2.0. In VideoPose2.0, roughly 30% of all lower arms are significantly foreshortened; see Figure 2 for a summary of the dataset variability.

In this work, we overcome these computational and modeling limitations using an ensemble of tractable sub-models which couple locations of body joints within and across frames using rich image-dependent cues. We address the problems of foreshortening and part scale variation by using joint centers (shoulders, elbows, wrists) as parts [13, 26, 19], instead of limbs. Each submodel in the ensemble tracks a single joint through time (e.g., left elbow), and also models the spatial arrangement of all joints in a single frame. Because of the tree structure of each submodel, we can perform efficient exact inference and use rich temporal features that depend on image appearance, e.g., color tracking and optical flow contours. The models are trained discriminatively using a large-margin loss, and we experiment with a range of inference techniques that introduce increasing coupling between models (and thus increasing computational cost). Intriguingly, we find that a highly efficient method of enforcing agreement on single variables that we introduced in [28] outperforms costly approximate inference using dual decomposition.

Our earlier work [28] first investigated an ensemble of models for tracking body parts, but with several key differences: the model is designed to simply prune away unlikely hypotheses for torso and shoulder locations, whereas in this work we focus on predicting the location of shoulders, elbows and hands—a significantly harder task. Furthermore, like other previous work, [28] shows only a small improvement over single-frame parsing, and likewise the only temporal cue used is geometric persistence.

We apply our motion parsing model on a new video dataset of highly varied and articulated poses from TV shows containing over 1,200 frames. We show significant quantitative and qualitative improvements over state-of-the-art single-frame pose estimation approaches (in particular, wrist and elbow accuracy improves by greater than 15%). In summary, the novel contributions of this work are: (1) an efficient ensemble of tree models for parsing human motion which covers a complex set of pairwise relationships and includes rich, image-dependent features; (2) stretchable 2D layout models, as opposed to the rigid parts typically used in pose estimation; (3) significant improvement over single frame parsing results (arguably a first in this setting) (4) a challenging new video dataset, VideoPose2.0, with a high degree of motion pose variability which shows limitations of state-of-the-art methods. The VideoPose2.0 dataset, code and videos are available at http://vision.grasp.upenn.edu/video.

2. Modeling

As discussed, due to the intractability of jointly tracking multiple objects through time, previous work has made limiting assumptions in both the representation of pose (i.e., the inability to model foreshortening or fine angular granularity), and the interactions between parts, which do not capture rich, image-dependent relationships. We address each of these issues in a principled framework.

2.1. Stretchable models of human pose

One of the biggest limitations of current models of human pose is the “cardboard people” assumption [10]: that body parts are rigid patches of fixed size (up to a global scale). In the Pictorial Structures (PS) framework [7], the human is represented as a collection of parts with fixed
lengths, which along with the position and angle of each joint, completely determines the pose of the person. Thus, wrists are always a fixed distance away from elbows in these models. This is a frequently violated assumption in realistic video, due to foreshortening and variation in body type. In the VideoPose2.0 dataset we introduce (Section 4), 30% of lower arms are significantly foreshortened.

Two joints > one limb: Rather than model human limbs as a position and orientation, we propose a model directly on the pixel coordinates of each joint. Although this choice introduces more variables (one for each joint rather than for each part), the state space for each variable is drastically reduced because we no longer need to reason over a finely discretized set of angles in the state space, and we can now implicitly represent nearly any angle between parts. In current PS implementations, this is a $24 \times 1283$ reduction in the state space of each variable. In addition to this, because the length of the limb is now determined implicitly by the pixel distance between neighboring joints, the model naturally lends itself to capturing large variability in the part length. Because of its ability to represent finely discretized limb length, we refer to this model as stretchable, in contrast to the typical rigid, rectangle-based representation.

One side-effect of switching from a limb-centric to joint-centric model of pose is that unary attributes of the limb-centric model are now pairwise attributes of a joint-centric model. Furthermore, pairwise attributes in a limb-centric model correspond to ternary attributes in a joint-centric model, which we do not use. However, in a standard PS model, pairwise attributes are only image-independent functions of geometry, whereas in our model, pairwise potentials all incorporate image information, giving us overall a more expressive model than standard PS.

2.2. Structured Ensemble of Stretchable Models

Ideally we want a model of human motion which captures important relationships between all correlated parts. This includes parts that are connected kinematically (e.g., left elbow, left wrist), parts that are left/right symmetric (e.g., left elbow, right elbow), and instantiations of the same part in consecutive frames (e.g., left elbow at time $t$, left elbow at time $t+1$). Clearly, modeling all these relationships together leads to cyclic dependencies within a single frame (due to the three symmetry edges) and between consecutive frames (due to the six tracking edges); see Figure 1, left.

However, in general, it is always possible to express the score of a given state assignment in a full, intractable model as the sum of scores under a set of tree sub-models that collectively cover every edge in the full model. This is the key insight that allows us to include all the rich relationships we desire: we decompose our model of all interesting relationships related to parsing human motion into an ensemble of submodels, all of which are trees (and therefore tractable). Each tree submodel is responsible for tracking a single joint through time and additionally models the corresponding set of pairwise interactions between joints in a single frame (Figure 1, right).

2.3. Formulation

Formally, we pose this problem as a structured prediction task, where the input $x$ is a video sequence of $t$ images and the output $y$ is a sequence of $P t$ variables, where $P$ is the number of parts (joint locations) included in the model. Each output $y_i$ is the 2-D coordinate of some part (defined in a $80 \times 80$ discretization of the pixel space) in some frame. We also use the shortcut notation $y_{ik} = \{y_i \mid y_i \text{ is in frame } t\}$ to index all $P$ joint variables in frame $t$.

Given a training set $S = \{(x^{(j)}, y^{(j)})\}^n_{j=1}$ of samples from a joint distribution, the standard supervised learning task is to learn a hypothesis $h: X \rightarrow Y$ that minimizes the expected loss $\mathbb{E}_S [\mathcal{L}(h(x^{(j)}), y^{(j)})]$ for some non-negative loss function $\mathcal{L}: Y \times Y \rightarrow \mathbb{R}^+$. The linear hypothesis class we consider is of the form $h(x) = \arg\max_y \theta(x, y)$, where the scoring function $\theta(x, y) \triangleq \theta^T f(x, y)$ is the inner product of a vector of parameters $\theta$ and a feature function $f: X \times Y \rightarrow \mathbb{R}^d$ mapping $(x, y)$ pairs to $d$ real-valued features. Let $G = (\mathcal{V}, \mathcal{E})$ be our full graphical pose model; we further assume that $f$ decomposes over the vertices $\mathcal{V}$ and edges $\mathcal{E}$, so that

$$\theta(x, y) = \sum_{(i, j) \in \mathcal{E}} \theta^T f_{ij}(x, y_i, y_j) + \sum_{i \in \mathcal{V}} \theta^T f_i(x, y_i). \tag{1}$$

Note that edges $(i, j)$ may connect variables between consecutive frames. Let there be $M$ tree sub-models, and $G_m = (\mathcal{V}, \mathcal{E}_m)$ be the sub-graph of $G$ corresponding to the $m$th one as in Figure 1, right. Then we decompose the score $\theta(x, y)$ into the sum of the scores of the $M$ constituent sub-models: $\theta(x, y) = \sum_{m=1}^M \theta_m(x, y)$, the score of the $m$th model is as in Equation 1, restricted to the edges $\mathcal{E}_m$, i.e. $\theta_m(x, y) = \sum_{(i, j) \in \mathcal{E}_m} \theta_{ij}^m f_{ij}(x, y_i, y_j) + \sum_{i \in \mathcal{V}} \theta_i^m f_i(x, y_i)$. Note that we do not couple parameters across different models $\theta_m$.

2.4. Structured Model Inference

We explore several methods for combining the $M$ independent models during test time to make a single final decision. The methods form a hierarchy of agreement criteria between the submodels: at one extreme, we enforce the constraint that all submodels must agree on the maximizing assignment to all variables, and at the other, inference is completely decoupled across submodels. Note that there is an inherent trade-off between the degree of agreement im-

---

1 The seminal work of Felzenswalb et al. [7] uses 10 discretized scales per part, but all modern implementations of PS use one fixed scale [21, 8, 1], partly due to its prohibitive increase in the state space.
posed on the models and the computational cost of the corresponding inference. We present our inference methods by order of decreasing agreement below.

**Full Agreement via Dual Decomposition.** A natural goal is to find the argmax decoding of joint locations throughout the entire sequence of frames using our original model in Eq. 1. However, solving the argmax decoding problem exactly is prohibitively expensive, due to the high treewidth of this cyclic graph. We use the method of Dual Decomposition (DD) [3, 11] to solve a linear programming relaxation of the decoding problem as follows. Observe that the argmax decoding problem of our full model can be decomposed into $M$ sub-problems if those problems are coupled through a global equality constraint:

$$\arg\max_{y, y'} \sum_{m=1}^{M} \theta_m(x, y_m) \text{ s.t. } y_m = y \quad (DD) \tag{2}$$

Although the optimization (2) is still intractable because of the integral constraints, optimizing the dual of (2) is always tractable (if we first drop the integrality requirement), but no longer guaranteed to return an optimal integral solution. We solve the dual problem with sub-gradient descent. Once complete agreement between all $y_m$ is reached, then inference has converged to the exact solution of (2) (although eventual convergence is not guaranteed.) In our experiments, if dual decomposition did not converge (after 500 iterations in practice, each requiring inference in all $M$ models), we used the maximum scoring primal variables found during any descent iteration.

**Single Frame Agreement.** An alternative to computing the MAP decoding of the entire sequence is to find the argmax solution while constraining model agreement to only a subset of the variables at a time. If we restrict our attention to the variables in a single time frame only, inference is considerably simpler. For each frame $t$, we solve

$$\arg\max_{y_t'} \sum_{m=1}^{M} \max_{y:y_m=y_t'} \theta_m(x, y) \quad (SF) \tag{3}$$

to obtain the joint configuration for that frame (remember $y_t'$ denotes the set of $P$ joint variables in frame $t$). In words, the inner maximization finds the highest scoring sequence $y$ subject to the constraint that the variables in frame $t$ are fixed at positions $y_t'$. The outer argmax is over all possibilities of single frame configurations $y_t'$. This extends the notion of a max-marginal of a variable (see [29]) to a max-marginal over a set of variables. This method requires first computing the max-product messages sent to variables in frame $t$, in each of the submodels. Finding the argmax decoding of $y_t$ is then equivalent to inference in a grid with $P$ variables (in our case $P = 6$ joints). This can be solved exactly by forming cliques of size 3, for which the message passing clique-tree forms a chain. In each clique, there are at most pairwise potentials, and since the state space of each part is relatively small ($|Y_p| \leq 500$), the $O(\sum_p |Y_p|^3)$ cost of this inference takes less than a second per frame in practice, and in our experiments took about as long as performing inference in all $M$ tree submodels (i.e., twice as slow overall).

**Single Variable Agreement.** We can further narrow our subset of interest down to a single variable at a time [28]. This is a weaker criteria for model agreement, but yields cheaper and simpler inference. This gives us the following inference problem for the $i$th variable:

$$\arg\max_{y_i'} \sum_{m=1}^{M} \max_{y:y_i'=y_i'} \theta_m(x, y) \quad (SV) \tag{4}$$

This can be solved by computing max-marginals for each model using standard forward-backward message passing, summing the $M$ max-marginal scores, and taking the highest scoring sum. Note that this is actually equal to MAP decoding in the full model when all sub-models agree on the argmax, which rarely occurs in practice. However, in [28] we showed empirically that this decoding is a useful approximation of the full MAP decoding prediction.

We also compared the above methods to predicting each joint using the single model in the ensemble that incorporated temporal dependencies for that specific part, which we call the **Independent** decoding scheme.

2.5. Learning

Although we enforce agreement during ensemble inference at test time, coupling inference during training for more than one variable is prohibitively expensive to use as part of a learning procedure. Thus we learn parameters $b_m$ using decoupled inference separately for each model. To learn parameters, we optimize a convex hinge-loss objective for each model separately. Let $\theta_m^{(i)} = \max_y \theta_m(x^{(i)}, y)$ and $y^{(i)}$ a corresponding maximizer, then the learning problem is

$$\min_{\theta_m} \frac{\lambda}{2} \|\theta_m\|^2 + \frac{1}{n} \sum_{j=1}^{n} \left[ \theta_m^{(i)} + 1 \left( y^{(j)} \neq y^{(i)} \right) - \theta_m(x^{(j)}, y^{(j)}) \right]_+$$

where $[.]_+ = \max(0, \cdot)$. We optimize this problem via stochastic sub-gradient descent, and we choose $\lambda$ and the number of training epochs using a held-out development set to minimize error.

3. Features

As described in Section 2, our model represents human pose and motion with an ensemble of tree models
which capture relationships between different joint locations within a frame and the relationships between the same joint across time. Due to the tractable nature of our tree decomposition, and the sparse set of states provided by a single frame cascaded PS [21], we can afford to combine a variety of effective features for all unary joint and pairwise joint-joint relationships we wish to model. Figure 3 illustrates most of the features described here.

### 3.1. Single-frame features

**Geometry.** Unlike previous single-frame geometry features used in PS representations, we purposefully only include coarse geometric relationships into our model, and rely more heavily on image-based cues to estimate pose. This is inspired by the observation that state-of-the-art PS implementations learned on existing datasets tend to learn very rigid geometric priors upon which they rely heavily [25]. These models generalize poorly, especially to important application domains with a high degree of pose variation, such as action recognition.

In light of this, we use the following pairwise geometric features: (i) length of arms, (ii) unsigned difference in angle that the upper arm makes with respect to the vertical axis, (iii) difference in x-coordinate between left-right symmetric joints, from which our model can learn a type of repulsion behavior (e.g., the left shoulder should be far from the right shoulder) as well as a left-right order of parts (e.g., the left shoulder should be to the left of the right shoulder). These features are coarsely discretized into 10 bins. Note that we do not express features describing the angle of the lower arm, leaving it free to rotate and stretch. See Figure 3d.

**Color-based hand detector.** Detecting the hand location is an extremely useful cue in anchoring all joint locations. Unfortunately, traditional template-based part detectors such as Histogram of Gradient (HoG) detectors fail at this task due to hands’ high variability in appearance, pose, and motion blur. We instead learn a linear SVM filter on skin detection response maps computed from the publicly available code from [21] on the training data. This can be evaluated efficiently using convolution at test time. See Figure 3a.

**Contour support.** For each arm joint-pair, we measure its support from long contours extracted in the image as follows: we take the number of contour points that are roughly parallel to the joint-pair line segment (angle less than 12 degrees) and within a spatial support region (approximately 25% of the length of the average groundtruth limb). We then use as a feature the number of supporting contour points, normalized by the length of the limb hypothesis. Due to the sparse contour set, this feature can be computed extremely efficiently, by quickly discarding hypotheses whose endpoints are not near any contour. See Figure 3c.

**Color consistency.** To capture the fact that the color of pairs of joints is often similar due to clothing and/or skin color, we describe the color consistency of pairs of joints via the $\chi^2$-distance between color histograms obtained from small image patches (with side length 10% of image dimensions) extracted around each joint. See Figure 3b.

**Figure from flow.** We use several features based on dense

---

Figure 3. Overview of features described in Section 3. (a) Two discriminative hand detector filters from optical flow and skin color. (b) Quantized color is matched within a frame (comparing color distributions) and over time (comparing $L_2$-norm patch distance). (c) Limb hypotheses are scored based on alignment to nearby contours. (d) We use a few simple geometric features between joints in a frame, and joint persistence over time. (e) We form an estimate of foreground and background likelihood from dense optical flow.
optical flow from [14], which we compute between adjacent frames. We obtain a rough estimate of the foreground of each clip by assuming there are only 2 planes of motion: foreground (figure) and background. Given a detected person, we estimate the background motion by computing the median flow vector $\mu_{bg}$ outside the detected person bounding box, and not considering outlier flow with magnitude greater than the 75th percentile. We then subtract off $\mu_{bg}$ from the flow field and take the magnitude of the flow as an estimate of foreground likelihood, as in Figure 3e. We incorporate this as a unary feature for each joint, and a pairwise feature by computing the average sampled evenly along the line segment between joint pairs.

**Flow-based hand detector.** We exploit the fact that hands are often in motion (and naturally the fastest moving body part) by building a hand detector based on hand-shaped motion discontinuity. We extract motion discontinuities by computing the gradient magnitude of the flow field, and learn a linear filter via SVM using this motion discontinuity magnitude cue specific to hands, similar to the single frame hand detector based on skin-color likelihood maps. See Figure 3a, top row.

**Joint contour and flow support.** We include an additional contour feature restricting our single-frame contour feature to only count support from contour points that are consistent with a large magnitude optical flow discontinuity. This serves to reduce background clutter and restricts support to only contours that are salient both spatially and temporally.

**Single-frame features from [21].** We make use of several feature computations provided by the public implementation of Cascaded PS [21], a state-of-the-art single frame pose estimation model; see paper for details.

We incorporate HoG limb detectors as unary features for each shoulder and elbow joint by taking the max response of the detector over all possible angles. We also incorporate the detectors as pairwise terms for (shoulder, elbow) and (elbow, wrist) pairs for which we can index the detector at the appropriate angle. The image-adaptive discriminative color models of clothing and skin color are incorporated as unary features. Finally, we discretize our state space relative to the initial detected upper body into a 5x5 grid, and use membership in each grid cell as a feature. This is a particularly effective feature for the shoulders, whose detection-relative location is relatively peaked, but not very informative for elbows and wrist locations in our highly articulated dataset (summarized in Figure 2).

### 3.2. Between-frame features

**$L_0$-norm quantized color tracking.** We capture the persistence of appearance over time with a simple and effective patch-based color tracker for each joint. We jointly quantize each pair of consecutive images into a small number of color indices, using minimum variance quantization\(^2\). We then compare patches (of side length 10% of the image dimensions) around the joint in each frame using an $L_0$-norm (Hamming loss) distance function. Let $P_t$ and $P_{t+1}$ represent quantized color image patches around the joint in frame $t$ and $t + 1$, with $N$ pixels. Then the color tracking distance feature is $||P_t - P_{t+1}||_0 = \frac{1}{N} \sum_{(r,c)} 1 (P_t(r,c) = P_{t+1}(r,c))$ where $(r,c)$ indexes rows and columns in the patch. This patch distance is robust to pixel value fluctuations and outliers, and encourages the pixel structure to be similar, unlike in color histogram distance tracking methods (e.g., [18]).

**Geometry.** To express joint motion continuity, we use one simple geometric feature: the Euclidean distance between the joint in consecutive frames, discretized into 10 binary values (Figure 3d).

### 4. Experiments

**VideoPose2.0 dataset** We apply our framework to the challenging task of tracking upper and lower arms in video clips taken from the TV shows *Friends* and *Lost*. We chose to focus on these parts because the remaining upper body parts (head, torso) can be localized with near perfect accuracy with current methods given a detection window, whereas lower arm localization performance of state-of-the-arts methods is still quite poor [21]. Furthermore, most things we are interested in knowing about humans involve the hands—e.g., action recognition, gesture identification and object manipulation.

Clips in the dataset were hand-selected (before develop-

---

\(^2\)In practice we use MATLAB’s `rgb2ind()` using 32 colors.
ing our algorithm) to highlight natural settings where state-of-the-art methods fail: a highly varied (yet realistic) range of poses, rapid gesticulation, and a significant portion of frames (30%) with foreshortened lower arms. The dataset consists of 44 short clips, 2-3 seconds in length, with a total of 1,286 frames. We use 26 clips for training, recycle 1 training clip for a development set, and use 18 for testing. The dataset fixes global scale and translation of the person, as is typically assumed in order to avoid confounding detection errors with pose estimation errors.

**Implementation details.** Our chosen ensemble of tree models is shown in Figure 1. This collection of six models captures time persistence of each of the six joints, as well as left/right symmetric joint edges for left/right shoulder, elbows and wrists. This covers all reasonable connections we could conceive of modeling, and allows us to incorporate all features mentioned in Section 3.

As input to our method, we use potential shoulder and elbow locations generated by the coarse-to-fine cascade of [21] independently for each frame. This typically yields 300-500 possible shoulder and elbow locations on average. Finally, for each of the 24 discrete elbow orientations predicted by [21], we project possible wrist locations at 4 different lengths, chosen from the 5th, 25th, 50th, and 75th lower arm length quantiles on the training set. We then take the top 500 wrist locations scored according to the foreground color features for each frame (Section 3). The result is a sparse set of locations for each joint with high recall.

**Baseline methods.** We compare to two state-of-the-art baselines in single-frame pose estimation: (1) Eichner et al.’s PS implementation [6] which performs exhaustive inference in a single frame, and uses multiple re-parses using color. It is worth mentioning again here that the authors have found that incorporating temporal information into their model (geometric persistence of parts via loopy belief propagation) achieves worse results than this single-frame model. (2) Sapp et al.’s cascaded PS model [21] which is the best performing single-frame pose estimation method on the Buffy and PASCAL Stickmen datasets [6]. Note that we use the cascade architecture from [21] to obtain our initial set of state hypotheses for each part, but compare to the final level of that model which uses more powerful features and nonlinear classifiers. Both baseline implementations are publicly available on the web.

**Evaluation.** We evaluate by measuring the percentage of joints within a given matching distance. By varying the matching distance in a reasonable range (15-40 pixels, see Figure 6, inset), we obtain a recall curve for each joint. We apply median filtering to smooth all trajectories.

**Results.** Quantitative results are shown in Figure 5. All three methods we explore for inference outperform the two state-of-the-art single-frame methods by a significant margin, and perform comparably to each other. At the tight matching thresholds, we outperform the baseline on elbows by approximately 12% and on wrists by 10%. The absolute accuracy values are also a testament to the difficulty of the dataset: At the tightest matching criterion, we only correctly localize half the elbows. This suggests that there is plenty of future progress to be made on this dataset, and in this domain in general.

Furthermore, we found that the very fast, approximate decoding schemes (Independent and SV) require only a single inference pass were comparably accurate to more expensive methods (SF takes about $2 \times$ as long while we ran DD for $500 \times$ iterations, each requiring inference in each submodel)\(^3\). We found two important trends: (1) On average, completely decoupled inference (Independent) was consistently about 0.75-1.5% worse than the inference methods that aggregated information across models, and (2) solving partial agreement problems exactly (SV, SF) performed better than solving complete agreement approximately with Dual Decomposition.

**Ablative feature analysis.** To determine which features make the most impact in our system, we trained different models removing one feature type at a time from the complete model (“All features”). The results in Figure 6 (compare with the SV inference method) reveal several interesting trends: (1) Our complete model outperforms a model without temporal edges (“All except Time”) by 2-3%, confirming that our proposed model can exploit time information for better results. (2) Using all features performs the best in a high precision regime, but removing the hand filter features actually helps in the lower precision, higher recall. We suspect that the hand filters help localize well when they are correct, but can corrupt the signal with spurious responses as well. (3) By far the single most important feature is the estimate of foreground obtained via optical flow, which contributes nearly a 10% boost in performance.

---

\(^3\)Specifically, running inference in all 6 models sequentially takes about 16 seconds per 30 frame clip (trivial to parallelize) on a Intel Xeon CPU, E5450 @ 3.00GHz. SF inference takes an additional 19 seconds.
Figure 6. Ablative feature analysis, see Section 4. Inset: Visualization of the minimum (15 pixels) and maximum (40 pixels) matching criteria we report in our results.

5. Conclusion

We have presented a novel ensemble-based approach to address the problem of parsing human motion. We exploit the efficiency of our approach to design rich image-dependent features for all the important joint-joint relationships we wish to model. Finally, we use a simple max-marginal combination algorithm to significantly outperform single-frame parsing by a large margin, something previous work has failed to achieve.

Acknowledgements. The authors were partially supported by NSF Grant 0803256, ARL Cooperative Agreement W911NF-10-2-0016 and an NSF Graduate Research Fellowship for D. Weiss.

References