Pose Tracking by Efficiently Exploiting Global Features

Ratnesh Kumar
Placemeter, NY
ratnesh@placemeter.com

Dhruv Batra
Virginia Tech, VA
dbatra@vt.edu

Abstract

Typical pose tracking algorithms first obtain a set of plausible pose hypotheses in all image frames of a video and subsequently stitch compatible detections across time to form a pose-track. This approach to tracking is commonly termed tracking-by-detections, and has been very successful in other areas such as multiple object tracking, video segmentation using object proposals. Often models in this category can only incorporate local spatio-temporal evidence due to exponentially increased cost when using global information. Local spatio-temporal evidence can be ambiguous, thus leading to an inferior objective modeling. To deal with ambiguities in local information it is necessary to incorporate global information over multiple frames into a model.

Based on the recent advances in generating multiple solutions from a probabilistic model, we first generate multiple plausible pose-track hypotheses, and subsequently employ a mixture of local and global features to express the quality of these solutions with high fidelity. We perform extensive experiments and competitive results across varied datasets demonstrate the robustness of our approach.

1. Introduction and Motivation

Pose estimation in videos is one of the key computer vision problem and is often a required or desirable step for an action or activity recognition system [11]. As with many other computer vision problems, key challenges in this task include camera-sensor noise, diverse scene illumination and camera viewpoints. Owing to the importance of this task, several approaches exist in the computer vision literature. However the accuracy is far from being perfect and localization of key-joints *e.g*. elbows and wrists is still poor.

Most recent approaches for tracking and segmentation are Conditional Random Field (CRF) models. The optimal solution for a CRF model is referred as Maximum a Posteriori (MAP) solution. Typical CRF models for video segmentation and tracking incorporate *local* color and motion cues by linking CRF variables which lie on successive frames. These local cues from adjacent frames can be ambiguous due to reasons such as camera noise, low frame rate and so on. One way to deal with this local evidence ambiguity is to incorporate long range temporal connections between variables (of a CRF) that are far apart in time, thus utilizing *global* consistency based information regarding color or motion. Unfortunately, the presence of global terms typically makes the inference in such models intractable.

A desirable global affinity measure in tracking is the *clique color homogeneity* requiring all nodes of a possible tracking solution to be color consistent. Incorporating such clique affinity challenges the inference of a model and one has to resort to meta-heruristic approaches *e.g*. Tabu search in [36]. Curvature penalization is another higher order factor utilized in multi-target tracking approaches (*e.g*. [15, 18]) leading to deal with approximate inference [8].

A way to incorporate global information in a model is to generate multiple solutions from a simpler model and enforce the global terms only on these solutions. These multiple solutions provide options to an user to pick a solution that is the best solution for the real problem. Notice that this best may be very different from the original best solution computed by the model. For an automated task such as pose estimation or an image segmentation these varied solutions are required to be *ranked automatically*. Usually the number of diverse plausible solutions generated ($\sim 10^{1\sim3}$) is significantly smaller than the space of all possible solutions ($\sim 10^{10}$), and various global features can be seamlessly added to distinguish a good solution from the rest.

An important issue with vision models is the generalization error. Large scale studies such as [28], [13] have typically found that a MAP solution is often of poorer quality than the ground truth. Yadollahpour *et al*. [33] argue that such generalization issues can be mitigated by producing multiple solutions and not just a single solution.

Based on the above arguments about the need for global terms and tackling tractability issues for a CRF model, we view pose tracking problem as a two staged process: (i) In the first step we compute diverse pose-track hypotheses, and (ii) find the best pose-track from the above obtained pose-track hypotheses set, using apt local & global features.

2. Related Works

Owing to the importance of pose estimation for video processing and image analytics, there are many existing approaches in literature, *e.g*. [26, 24, 17, 9, 2, 35, 20]. Batra *et al*. [4] improve the *pose detection* of [20] by utilizing diverse solutions.

Pose Estimation in Videos: Pose Estimation approaches for a video, can be broadly divided into two categories: (1) articulated motion parsing, and (2) tracking-by-detections. Approaches under the first category jointly use spatial and temporal information to estimate a pose-track in a video. Tracking-by-detections is a two staged approach wherein pose detections are first computed for all (or a few) frames of a video. Subsequently temporal information (along with spatial cues) is used to find the most compatible set of poses across the video thus determining a pose-track.

Some of the popular motion parsing approaches to estimate pose in a video are [40, 27, 25, 6, 9, 16, 10].

Pose tracking-by-detections: These models naturally induce a network flow architecture, wherein the best track is determined by finding the maximum flow that can be pushed from the source to a sink. The work by [21] establishes *approximate* equivalence of *network flow models* relating to multiple object tracking to the chain-CRF models which are typically solved by dynamic programming (Viterbi). This equivalence is approximate in the sense that inference on a *Directed Acyclic Graph* (DAG) model is suboptimal if the number of objects is > 1. Multiple object tracking has a long history of employing network flow models, and some notable works are by [12, 3, 38, 5].

Some of the recent approaches in pose tracking-by-detections which employ a chain CRF model are by [20, 23, 7]. Park *et al*. [20] generates M-best pose hypotheses for each frame, and then use a chain CRF to stitch the most plausible pose track. The tracking model of Cherian *et al*. [7] is similar to Park *et al*. [20]. Cherian *et al*. [7] tracks body parts individually, and imposes regularization along the limbs connecting body parts.

Ramakrishna *et al*. [23] use a network flow model wherein part-detection hypotheses are the nodes and appropriate edges (upto order 3) are incorporated in their model. A pose track is computed in a top-down fashion wherein they first compute the track for the head joint. Subsequently proposals for the next symmetric pair (*e.g*. hands) are computed by conditioning on the tracked locations of the previous found parent track. The authors here manage occlusion explicitly in the state space for a node. Ramannan *et al*. [25] first find a frame with an easily detectable canonical pose which is used to build an appearance model for a person to aid tracking for the rest of the video. A shortcoming of such approach is that canonical pose is often hard to find.

The closest approach to ours is *tracking-by-selection* by Tokola *et al*. [30]. Here the authors first generate a set of plausible tracks using detection hypothesis from [27]. These multiple pose-tracks are re-ranked using the modified inference model of [27]. The authors express the quality of a pose track by measuring color coherency along a track and reduce this reward for a track by a factor determined by the max-displacement for a track. There are following major differences w.r.t. our proposed approach:

- The work by [30] compute pose tracks by sampling per frame detection hypotheses, and no explicit criterion to measure the difference between two tracks is incorporated. Without an explicit diversity function among track, the solutions can become very similar to each other [4]. On the other hand, our work utilizes the DivMBest algorithm [4], which provides a formal way to compute multiple *diverse* tracks from a tracking model. DivMBest has a single degree of freedom to generate multiple solutions (while for the heuristic approach in [30] has 4 parameters per body-joint), and is fairly robust and need not require any training data.
- The criterions used to express the quality of multiple solutions by [30] can only correct mistakes which happen due to poor color consistency. A better feature pool *e.g*. long term motion consistency, is required at this step to deal with failure in capturing color coherency. Ablation study in Table 4 affirms that criterions in our re-ranker are quintessential and are complementary to one another.

Some noteworthy works in multiple-person tracking exploiting multiple solutions are by [34, 32].

With respect to the literature our approach presents the following advantages:

- A novel re-ranker with an apt combination of local & global features to rank pose-track hypotheses for a better solution *vs*. MAP solution, w.r.t. the ground truth.
- Utilizing the DivMBest framework, we provide a simple yet robust approach to generating *diverse* pose-track hypotheses. To the best of our knowledge this is the first application of DivMBest to generate diverse pose-tracking solutions.

The rest of the paper is organised as: in the following section we describe a typical chain CRF model which serves as the backbone for many pose tracking-by-detections approaches. An overview of the DivMBest framework to be applied to a chain CRF tracking model is provided in section 4. Subsequent sections will elaborate on various score functions used to quantify pose-tracks, followed by the experimental results in section 7.

3. Chain CRF for tracking-by-detections

A pose estimator (detector) is used to obtain a plausible set of pose hypotheses for all frames of a video batch. A

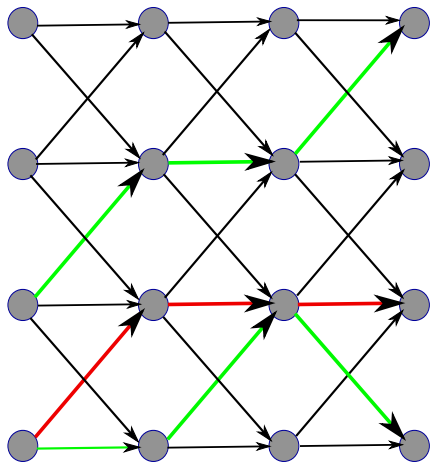

Figure 1. Typical *Chain CRF* model used in tracking. Circled nodes denote pose hypotheses and each column a time frame. Red colored edges indicate a MAP solution. To generate more solutions from the model, we constrain (softly) the previously found solutions by restricting the amount of flow through these paths. Green colored edges indicate two solutions obtained by constraining the solutions found earlier than the green colored solutions.

graph $G = (\mathcal{V}, \mathcal{E})$ is then constructed wherein the vertices $\mathcal{V}$ indicates set of pose proposals across all frames. Suitable weighted edges are incorporated between every pair of temporally successive pose hypotheses, (*c.f.* Figure 1). Let us denote,

$\mathcal{Y}$	:	Set of all possible tracks.
u	:	A node of the graph, *i.e.* a pose detection.
$\mathcal{V}$	:	Set of all possible nodes u.
y_u	:	Label of the node u, to be found.

A labeling $\mathbf{y} = (y_u)_{u \in \mathcal{V}} \in \mathcal{Y}$ defines a plausible pose track.

The optimum solution of this model is the optimum of:

$$\underset{\mathbf{y} \in \mathcal{Y}}{\text{argmax}} \quad \sum_{u \in V} \theta_u(y_u) + \sum_{(u,v) \in \mathcal{E}} \theta_{u,v}(y_u, y_v), \qquad (1)$$

where θ_u stands for unaries and $\theta_{u,v}$ corresponds to the edge weights encoding similarity between nodes. An exact inference can be performed on this model with the Viterbi algorithm (Dynamic Programming). This model (and its variants) is used in many temporal analytic problems, *e.g.* object tracking. With some modifications it can be adapted to video segmentation using object proposals [37].

State-of-the-art pose tracking models such as [20], [7] are based on the above chain CRF model for pose-tracking. We use the work by [7] as *base model* for extracting multiple solutions. Our approach is naturally applicable to any CRF-based model. We chose [7] due to it's competitive performance and source code availability. Cherian *et al.* [7] compute pose-tracks for each joints in a top down fashion. Firstly track for head is detected and subsequently (conditioning on earlier found tracks) the tracks for the lower joints are computed iteratively. Each track computation in this model is a chain CRF.

4. Multiple diverse pose tracks

We compute diverse tracks (solutions) in an iterative manner. Similarly to [4] we call our M diverse pose-tracks as **DivMBest** pose-tracks. First the best solution *i.e.* MAP is computed by optimizing criterion (1). The next best solution is found by constraining the MAP solution. We constrain a path (solution) by decreasing the unaries so as to make the next solution to be different from the current solution. A solution indicates the full set of tracks for each body joint. Hence we have M diverse tracks for each body-joints.

Let $\mathbf{y}^1$ be the MAP, $\mathbf{y}^2$ be the second solution and so on with $\mathbf{y}^M$ be the $\mathbf{M}^{th}$ best solution. At each step, the next best solution is defined as the highest scoring state with a minimum degree of dis-similarity as measured under a function $\Delta(\cdot, \cdot)$:

$$\mathbf{y}^M = \underset{\mathbf{y} \in \mathcal{Y}}{\text{argmax}} \quad \sum_{u \in \mathcal{V}} \theta_u(y_u) + \sum_{(u,v) \in \mathcal{E}} \theta_{uv}(y_u, y_v) \qquad (2a)$$

$$s.t. \qquad \Delta(\mathbf{y}, \mathbf{y}^m) \geq k_m \quad \forall m \in [M-1]. \qquad (2b)$$

In general, this problem is NP-hard and Batra *et al.* [4] proposed to use the Lagrangian relaxation formed by dualizing the dissimilarity constraints $\Delta(\mathbf{y}, \mathbf{y}^m) \geq k_m$:

$$f(\boldsymbol{\lambda}) = \max_{\mathbf{y} \in \mathcal{Y}} \quad S_\Delta(\mathbf{y}) \doteq \sum_{u \in \mathcal{V}} \theta_u(y_u) + \sum_{(u,v) \in \mathcal{E}} \theta_{uv}(y_u, y_v) + \sum_{m=1}^{M-1} \lambda_m \left(\Delta(\mathbf{y}, \mathbf{y}^m) - k_m\right). \qquad (3)$$

Here $\boldsymbol{\lambda} = \{\lambda_m \mid m \in [M-1]\}$ is the set of Lagrange multipliers, which determine the weight of the penalty imposed for violating the diversity constraints.

We use Hamming diversity, *i.e.* $\Delta(\mathbf{y}, \mathbf{y}^m) = \sum_{u \in \mathcal{V}} [\![y_u \neq y_u^m]\!]$, where $[\![\cdot]\!]$ is 1 if the input condition is true, and 0 otherwise. This function counts the number of nodes that are labeled differently between two solutions. For Hamming dissimilarity, the Δ-augmented scoring function (3) can be written as:

$$S_\Delta(\mathbf{y}) = \sum_{u \in \mathcal{V}} \underbrace{\left(\boldsymbol{\theta}_u(y_u) + \sum_{m=1}^{M-1} \lambda_m [\![y_u \neq y_u^m]\!] \right)}_{\text{Perturbed Unary Score}} \qquad (4)$$

Following [4] we use a single λ parameter for all m.

5. Expressing the quality of a trajectory

From a base tracking model, we generate M diverse pose tracks and the subsequent aim is to estimate the best track

for each body joint. A pose detection for an image is defined by a set of bounding boxes for corresponding body-joints. Hence a **joint-track** (Ω), corresponds to a sequence of boxes (*c.f.* Figure 2) for a body part *e.g.* elbows, wrists. From M diverse pose-track hypotheses we obtain a set of

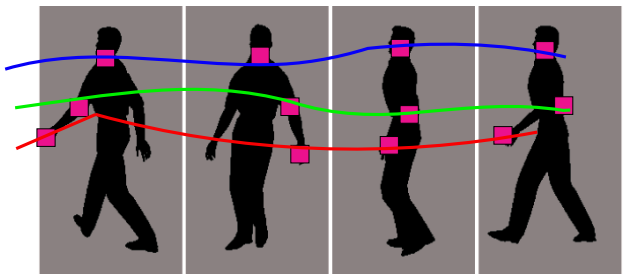

Figure 2. **Pose tracks**: A pose detection hypothesis on an image frame consists of bounding boxes for body joints such as shoulders, elbows and wrists. Figure shows detection joints for one side of the body, colored in Magenta. Tracks for these joints are shown in red, green, and blue. For M diverse pose tracks, we have M tracks for each body joint. Figure adapted from [22].

M tracks for each body joint.

We now describe the following relevant criteria to express the quality of a *joint-track* (Ω):

(C_1) Color similarity: The color of all boxes comprising a trajectory should be similar to one another.

(C_2) Long term motion affinity: Long term motion of nodes (*i.e.* bounding boxes) along a joint trajectory should be compatible w.r.t. one another.

(C_3) Temporal smoothness: A trajectory should be as smooth as possible in time.

(C_4) Motion boundary vicinity: Moving body parts such as wrists should be as close to a motion boundary as possible.

Incorporating **criterion** C_1 in a tracking model induces a maximum weighted clique affinity wherein weights correspond to the color similarity between nodes. This leads to a problem of finding maximal weighted cliques in a graph, which is a tough problem (NP-Hard) and to date no known algorithm exist which can guarantee the solution quality for this problem [39]. Approaches such as [36], [29] have looked into solving clique problems using heuristic optimization techniques which have no stability guarantees for output, and are typically run many times for obtaining a usable solution. However since the diverse solution space is much smaller as compared to the clique space in the base model, we can employ global clique criterion and find solutions better than MAP in a painless manner.

For our purpose to evaluate the diverse solutions for criterion C_1, we compute χ^2-distance between color histograms for *every* pair of boxes along a trajectory Ω. The color similarity $H(\Omega)$ of a trajectory Ω is:

$$H(\Omega) \;=\; \sum_{(i,j)\in E} \chi^2(h(i), h(j)), \tag{5}$$

where $h(i)$ indicates color histogram for a box i of a joint track, E is the set of all possible pairwise edges between boxes belonging to the trajectory.

Higher the value of $H(\Omega)$, lower is the track reliability. Hence we use the following function to make $H(\Omega)$ monotonically increasing w.r.t. quality of a trajectory:

$$H(\Omega) \;=\; exp(-H(\Omega)/\delta_1). \tag{6}$$

Notice that computing $H(\Omega)$ only asks for $\binom{30}{2}$ *i.e.* 435 summing operations for a track of length 30 frames. On the contrary, attempting to evaluate this global consistency from the base model (for 300 detections per frame) increases the problem complexity exponentially to $O((300 \times 30)^{30})$, and therefore would be extremely difficult to solve.

Criterion C_2 asks for estimating long term motion compatibility between nodes (boxes) along a trajectory. In order to estimate long term motion compatibility we use point-tracks[1] based affinity (*c.f.* Figure 3). Higher the number of point tracks shared across two boxes lying far apart, higher is their compatibility and hence are more probable to belong to the same trajectory.

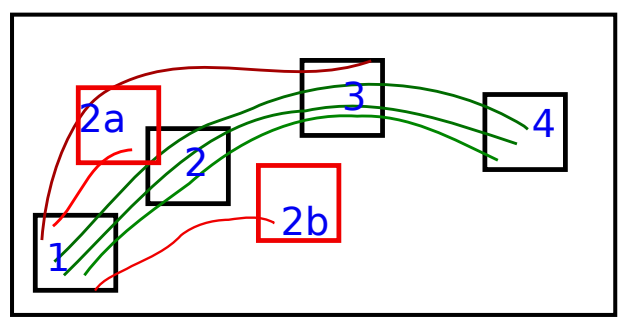

Figure 3. Illustration of point tracks based *long term* motion-compatibility ($MT(\Omega)$): Each box represent a joint location. Numbers inside boxes indicate their identities and frame locations, with frame 2 having 3 detections. Red boxes and point-tracks are incorrect detections and point-tracks. Criteria (7) will prefer the set of *black* boxes (sequence 1-2-3-4) than a set containing *red* boxes (*e.g.* sequence 1-2a-3-4 or 1-2b-3-4) due to higher number of shared point-tracks for the box-sequence 1-2-3-4.

In order to robustly estimate a temporally long term affinity/compatibility among boxes (or nodes), removal of background point tracks is of utmost importance. To this

[1] We explicitly add the word *point* before tracks wherever necessary to note *point-tracks*.

end we subtract the dominant motion of a scene (computed by [19]), and subsequently remove the static point tracks.

Long term motion compatibility $MT(\Omega)$ for a joint-track Ω, is then defined by:

$$MT(\Omega) \;=\; \sum_{pt \in (i,j)} l(pt), \tag{7}$$

where pt indicates point track and $l(pt)$ denotes the length of a point track (i and j, indicate boxes belonging to a track). The intuition for including $l(pt)$ instead of simply counting the number of common point tracks is that a longer length point track is more reliable and hence this factor gives increasing weightage to boxes that lie apart in time and are connected by long point tracks.

Notice that incorporating MT into a tracking-by-detection model, like H will ask for edge connections of the same order as the number of frames in a video batch, and hence would then be extremely difficult to solve.

Criterion C_3 prefers smooth trajectories and is measured by considering acceleration (or curvature) along a trajectory. Computing acceleration requires three consecutive nodes (boxes) along a trajectory. Incorporation of acceleration penalization into a base tracking-by-detection model (by adding edges of order three) sacrifices good properties of typical data association models employed in tracking-by-detection approaches [8], leading to approximate inference.

For a joint-track Ω we compute the centroid of bounding box in every frame, and the *laplacian* of centroids of three successive joint locations then expresses the acceleration (*c.f.* equation (8)) at a time frame t:

$$A(i,j,k;t) \;=\; \|p_j + p_k - 2p_i\|\,, \tag{8}$$

where i, j, k correspond to three temporally successive detections with i located at frame t, and j, k are located at $t-1$ and $t+1$ respectively. p_i indicates centroid of a box i.

A camera motion could induce jumps in the perceived motion, hence for a reliable acceleration computation we subtract the dominant image motion (as done for criterion C_2) from the centroids (p_i).

Acceleration for the full trajectory, $TS(\Omega)$ is then determined by summing over the acceleration at all time frames:

$$TS(\Omega) \;=\; \sum_{t} A(i,j,k;t). \tag{9}$$

Higher the value of TS for a track, lower is its reliability. Hence we use the following function to make TS monotonically increasing w.r.t. quality of a trajectory:

$$TS(\Omega) \;=\; exp(-TS(\Omega)/\delta_2). \tag{10}$$

Criterion C_4 asks for closeness of motion boundaries and moving part locations such as wrists. To this end we compute the *spatial* gradient of optical flow for all frames of a video, and sum over these gradients on the points along a joint-track Ω. Figure 4 shows sample motion boundary on frames from the VideoPose dataset [31], and demonstrates the importance of motion boundary for joints such as elbows and wrists. The motion boundary vicinity, $MB(\Omega)$ is expressed as:

$$MB(\Omega) \;=\; \sum_{i} \sum_{r \in i} \nabla ||F(r)||, \tag{11}$$

where $\nabla ||F(r)||$ stands for gradient of optical flow magnitude at spatial location r. In practice $MB(\Omega)$ is useful for rapidly moving joints *e.g.* wrists, and is unnecessary for shoulders.

Notice that unlike the above criteria (C_1, C_2 and C_3) which are computationally prohibitive if incorporated into the base tracking model, $MB(\Omega)$ can be incorporated into the base model (criteria (1)) as a unary term. However this will require re-training the original model parameters and hyperparameters. This outlines a very important aspect of diverse-solutions, *i.e.* incorporation of new features into a model without requiring to re-train the base model.

Unified score for a pose track

Summing up all the quantities discussed above, we obtain the following score ($\mathcal{S}$) for a joint track (Ω):

$$\mathcal{S}(\Omega) \;=\; \alpha H(\Omega) + \beta TS(\Omega) + \gamma MT(\Omega) + \zeta MB(\Omega). \tag{12}$$

Notice that none of the above scores measure the quality of a pose and displacements, hence we include the MAP score (measuring unaries, pairwise and skeleton regularization) of a joint-track into $\mathcal{S}(\hat{\Omega})$:

$$\mathcal{S}(\hat{\Omega}) \;=\; \mathcal{S}(\Omega) + MAP(\Omega). \tag{13}$$

Selecting the best track: The best joint-track $\bar{\Omega}$ is the track of highest score $\mathcal{S}$ in equation (13):

$$\bar{\Omega} \;=\; \underset{\Omega}{\operatorname{argmax}}\, \mathcal{S}(\Omega). \tag{14}$$

6. Implementation

We use the recent model and implementation of Cherian *et al.* [7] to compute diverse pose-track hypotheses. This implementation has two components: (i) An image based pose estimator which utilizes optical flow to provide multiple pose estimates (detections) per frame. The parameters of this model is trained on the FLIC dataset which comprises 4500 upper body annotated poses. (ii) The second component provides a pose track for each body part/joint using a chain CRF variant. All our experiments use the default hyperparameter setting provided by the authors.

Note that the tracking and detection parameters from [7] trained on VideoPose dataset are not publicly available.

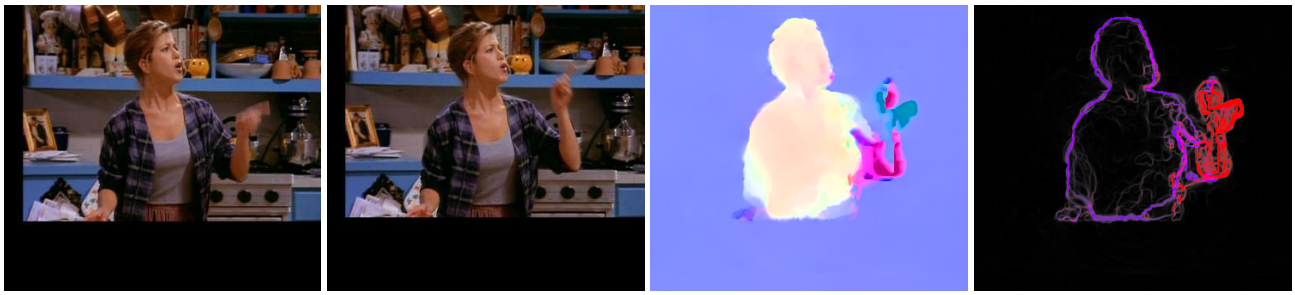

Figure 4. First two images show sample successive frames from the VideoPose dataset. Third and fourth images display optical *flow* and *gradient of flow magnitude* respectively. It can be observed that gradient of flow is a good indicator of fast moving keyjoints such as wrists.

Thus, we use the same default set of parameters in all our experiments.

We compute $M = 1000$ diverse pose-tracks by perturbing unaries (*c.f.* equation (4)). The parameter λ which determine the extent of perturbation for unaries is set empirically by grid search to 3 by using the train set of VideoPose dataset. A small perturbation on do not change the results and we reach similar accuracy over $M = 1000$ tracks. The parameters in equation (12) were set empirically by coarse grid search using the train set of the VideoPose dataset.

7. Experiments

For our evaluation purposes we use two different datasets: VideoPose and Poses in the wild.

VideoPose, VP: This dataset is introduced by [31] and has two versions. The two versions differ in their frame rates. We test our approach on the high frame rate version.

Poses in the wild, PIW: PIW is a challenging dataset introduced by [7] comprising many outdoor scenes, encompassing close to 900 frames for 30 sequences (30 frames per sequence). Most videos in this dataset are of low frame rate and are not a representative of the current day camera capture frame rate. In some videos more than 50 frames are skipped causing large displacements of the order of one-tenth of the video size. Therefore for our evaluation purposes we *remove* 13 videos from this dataset including the videos wherein the MAP and *Oracle* accuracy is close to zero, because no re-ranking based method can achieve non-zero performance on such datasets. More details for videos used can be found at [1].

Evaluation Metric: We use *percentage of keyjoint localization* accuracy for evaluation purposes. For a pose estimate, it measures percentage of keyjoints localized within a distance threshold (15 pixels) w.r.t. the ground truth.

Max vs. mean localization accuracy: In a few earlier approaches, the focus was on detecting the best symmetric parts, and approaches such as [7] report the maximum accuracy of the best localized symmetric part. This protocol has an obvious shortcoming: If a pose estimator (or a tracker) mis-detects one hand completely and detect another hand with full precision (*e.g.* first row in Figure 5), then the algorithm will report a 100% accuracy, which can lead to a poor assessment about a pose tracker. For our evaluations we provide the *mean localization error* of symmetric keyjoints.

Results and Discussion

There are two aspects of our approach: (i) Generation of diverse pose-track hypotheses, and (ii) Assigning scores to pose-track hypotheses in order to re-rank them.

Dataset	Shoulders	Elbows	Wrists
Oracle pose-track accuracy			
VideoPose	88.0	56.1	54.2
PIW	95.1	84.2	77.0
Oracle pose-detection accuracy			
VideoPose	99.1	79.5	77.5
PIW	90.6	91.1	82.2

Table 1. **Oracle accuracy**: *Oracle pose-track accuracy* is the accuracy when an Oracle selects the best track, and *Oracle pose proposal accuracy* is when oracle selects the best proposal per frame.

In order to evaluate the pose-track hypotheses, we estimate the best trajectory for each joint using the ground truth and call it *Oracle pose-track accuracy*. Table 1 shows the oracle pose-track accuracy for both datasets. Since we use detections on each image frame (tracking-by-detections), the oracle pose-track can only be as good as picking the best pose estimate per frame. The Oracle pose-tracker achieves better accuracy than best accuracy reported in earlier works: on VideoPose the state-of-the-art is atleast 10 points lower than the oracle pose-track accuracy for wrists, while on PIW the oracle pose-track accuracy is close to 20 points higher than the earlier best reported accuracy.

The accuracy of a pose tracker, if an oracle picks the *best pose detection* per frame is shown in Table 1. By design this accuracy will be better than the oracle pose-track accuracy.

VideoPose: Comparative results for the VP dataset are shown in Table 2. Our approach performs better than tracking-by-selection approach of [30] for the localization of wrists, and significantly improves over the recently published [7].

VideoPose (VP)		
Method	**Elbows**	**Wrists**
MAP (Cherian *et al*. [7])	38.8	27.7
Tokola *et al*. [30]	49.0	37.0
Zuffi *et al*. [40]	52.0	42.0
Sapp *et al*. [27]	48.0	30.0
Proposed	43.8	40.1

Table 2. Results on the VideoPose dataset. We significantly improve the MAP accuracy using *diverse* pose-track hypotheses, and an under-performing model now provides competitive accuracy with the state-of-the-art.

We improve the MAP scores on wrists by 44.5%; while [30] improves their base-machinery by 23.3% (wrists).

PIW		
Method	**Elbows**	**Wrists**
MAP (Cherian *et al*. [7])	57.2	54.7
Proposed	63.6	58.6

Table 3. Results on PIW. We significantly improve over the MAP.

PIW: The results and comparison with MAP [7] are shown in Table 3. We improve on MAP by a significant amount on this challenging dataset. Figures 5 and 6 show some qualitative comparison w.r.t. MAP (*c.f*. captions for details).

Evaluating components of our scoring function			
Dataset	**Component**	**Elbows**	**Wrists**
VP	Motion Compatibility	40.0	33.0
	Motion Boundary	42.6	35.8
	Temporal Smoothness	35.2	18.0
	Color Similarity	40.0	13,1
	Score from (13)	**43.8**	**40.1**
PIW	Motion Compatibility	49.0	50.0
	Motion Boundary	42.2	42.6
	Temporal Smoothness	57.0	38.0
	Color Similarity	43.2	12.6
	Score from (13)	**63.6**	**58.6**

Table 4. Ablation study of score function components. This table presents results of our approach if only one component of the score function in equation (12) is used. **VP** stands for VideoPose dataset.

Ablation studies[2] In order to evaluate the components of joint-track scoring function (*c.f*. equation (12)), we present results in Table 4 by using a single component of the score function. The general notion we get from this is that all the features used are complementary to one another, and hence are essential for obtaining a good pose-track. A color coherency based re-ranker (*e.g*. in [30]) would alone be inefficient in re-ranking solutions.

Motion boundary and point-tracks based motion compatibility appears very important and score much higher than other components for wrists. Motion boundaries are local in nature and can be susceptible to frame-by-frame noise, while longer point-tracks are more robust, but at the same time point-tracks (thus motion compatibility) are more dependent on the quality of motion compensation used, as background point-tracks (if not removed by motion compensation) will increase the score for spurious detections on the background. In most videos the persons are moving their hands rapidly and thus motion gradient offer useful information about wrist localization.

Color similarity works better for elbows than wrists, and our accuracies are closer to the elbow-oracle when elbows are visible. Wrists often have a wider span and speed of movement, and detections on wrists are comparatively noisier containing many pixels from the background. Therefore motion information appears vital for writs tracking.

Temporal smoothness achieves good results on both datasets underlining the importance of curvature penalization for assessing the quality of a trajectory.

[2]Detailed results for each video can be found at [1].

8. Conclusion

We proposed a pose-tracking by track-selection approach, wherein an explicit diversity criterion is incorporated into a base model to compute multiple solutions. These solutions are re-ranked by evaluating global consistency of relevant features, which are otherwise difficult to incorporate in the base model due to heavy tractability issues. With the proposed workflow we improve the accuracy of the base model by a significant margin and turn an under-performing model into a state-of-the-art model.

In future we will investigate the possibility of formulating the problem of computing diverse solutions for a tracking problem as labelings of a *single* energy minimization problem (*e.g*. [14]).

Another very interesting prospect is to design a suitable optimizer in diverse solution space so as to select *reliable parts* of solutions (instead of using a full solution) to construct an output (re-ranked) trajectory.

Acknowledgement: This work was partially supported by the following grants to **DB**: a National Science Foundation CAREER award, an Army Research Office YIP award, and an Office of Naval Research grant. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the U.S. Government or any sponsor.

References

[1] https://sites.google.com/site/kumarratnesh198521/home/.

[2] M. Andriluka, S. Roth, and B. Schiele. Pictorial structures revisited: People detection and articulated pose estimation. *CVPR'09*.

Figure 5. **First Row** displays (*colored skeletons*) the MAP solution of [7]. Notice the collapsing of one hand's detection (shoulder, elbow, wrist) onto another. **Second Row** shows our solution which provides better localization than MAP. **Third Row** shows Oracle solution. A color consistency re-ranker chose a solution very similar to MAP, underlining a need for features from motion.

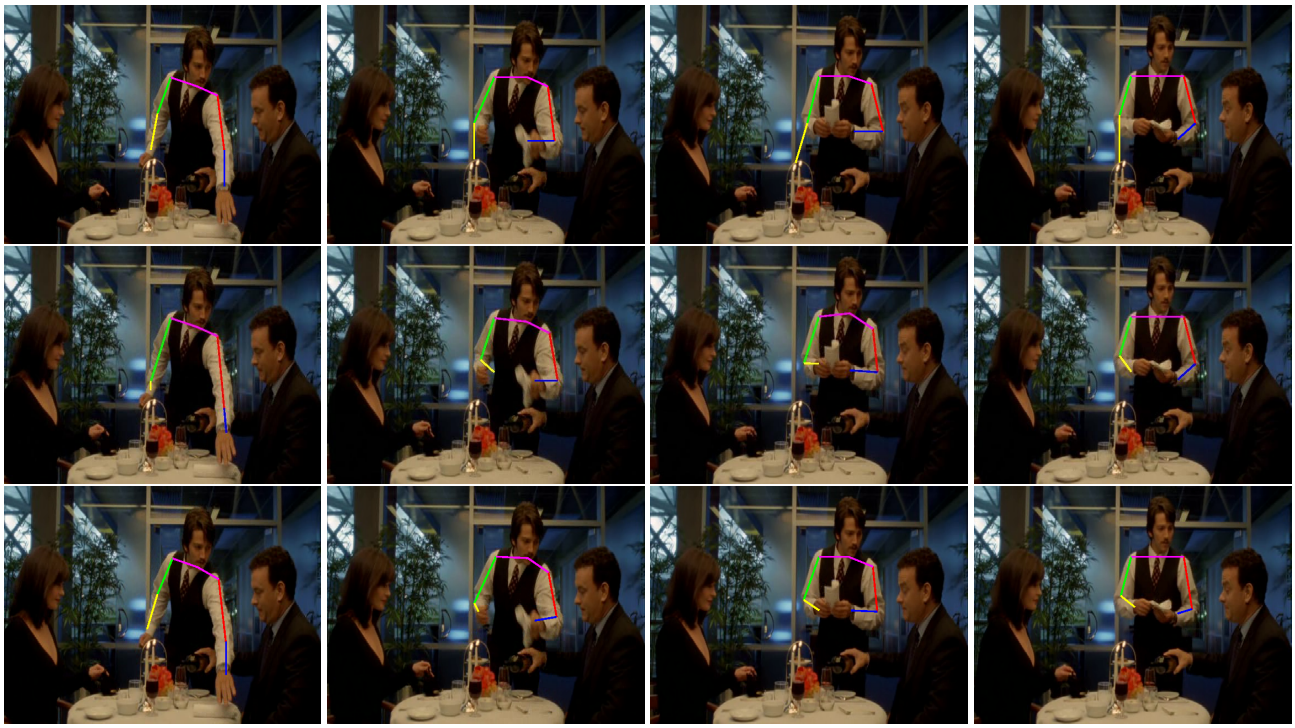

Figure 6. **First Row** displays (*colored skeletons*) MAP solution from [7]. **Second Row** shows our solution (scored very close to the oracle). **Third Row** displays the oracle pose-track. Notice that elbows in MAP are confused with the background. Thanks to our scoring function's long term motion compatibility and smoothness factors we are able to reach Oracle accuracy.

[3] A. Andriyenko and K. Schindler. Globally optimal multi-target tracking on a hexagonal lattice. In *ECCV'10*.

[4] D. Batra, P. Yadollahpour, A. Guzman-Rivera, and G. Shakhnarovich. Diverse M-best solutions in Markov random fields. In *ECCV '12*.

[5] J. Berclaz, F. Fleuret, E. Türetken, and P. Fua. Multiple Object Tracking using K-Shortest Paths Optimization. *TPAMI'11*.

[6] P. Buehler, M. Everingham, D. P. Huttenlocher, and A. Zisserman. Upper Body Detection and Tracking in Extended Signing Sequences. *IJCV'11*.

[7] A. Cherian, J. Mairal, K. Alahari, and C. Schmid. Mixing Body-Part Sequences for Human Pose Estimation. In *CVPR'14*.

[8] R. T. Collins. Multitarget data association with higher-order motion models. In *CVPR'12*.

[9] V. F, M. Martin-Jimenezs, and A. Zisserman. Progressive search space reduction for human pose estimation. In *CVPR'08*.

[10] K. Fragkiadaki, H. Hu, and J. Shi. Pose from Flow and Flow from Pose. In *CVPR'13*.

[11] H. Jhuang, J. Gall, S. Zuffi, C. Schmid, and M. J. Black. Towards understanding action recognition. In *ICCV'13*.

[12] H. Jiang, S. Fels, and J. J. Little. A Linear Programming Approach for Multiple Object Tracking. In *CVPR'07*.

[13] J. H. Kappes, B. Andres, F. Hamprecht, C. Schnorr, S. Nowozin, D. Batra, S. Kim, B. X. Kausler, J. Lellmann, N. Komodakis, and C. Rother. A Comparative Study of Modern Inference Techniques for Discrete Energy Minimization Problems. In *CVPR'13*.

[14] A. Kirillov, B. Savchynskyy, D. Schlesinger, D. Vetrov, and C. Rother. Inferring M-Best Diverse Labelings in a Single One. In *ICCV'15*.

[15] R. Kumar, G. Charpiat, and M. Thonnat. Multiple Object Tracking by Efficient Graph Partitioning. *ACCV'14*.

[16] T. Lim, S. Hong, B. Han, and J. H. Han. Joint Segmentation and Pose Tracking of Human in Natural Videos. *ICCV'13*.

[17] K. Mikolajczyk, C. Schmid, and A. Zisserman. Human detection based on a probabilistic assembly of robust part detectors. In *ECCV'04*.

[18] A. Milan, K. Schindler, and S. Roth. Detection-and Trajectory-Level Exclusion in Multiple Object Tracking. In *CVPR*, number June, 2013.

[19] J. Odobez and P. Bouthemy. Robust multiresolution estimation of parametric motion models. *Journal of visual communication and image Representation*, 1995.

[20] D. Park and D. Ramanan. N-best maximal decoders for part models. In *ICCV '11*.

[21] H. Pirsiavash, D. Ramanan, and C. C. Fowlkes. Globally-optimal greedy algorithms for tracking a variable number of objects. *CVPR'11*.

[22] S. Prince. *Computer Vision: Models Learning and Inference*. Cambridge University Press, 2012.

[23] V. Ramakrishna, T. Kanade, and Y. Sheikh. Tracking human pose by tracking symmetric parts. In *CVPR'13*.

[24] D. Ramanan. Learning to parse images of articulated bodies. In *NIPS'09*.

[25] D. Ramanan, D. A. Forsyth, and A. Zisserman. Strike a pose: Tracking people by finding stylized poses. In *CVPR '05*.

[26] R. Ronfard, C. Schmid, and B. Triggs. Learning to Parse Pictures of People. In *ECCV'02*.

[27] B. Sapp, D. Weiss, and B. Taskar. Parsing human motion with stretchable models. In *CVPR'11*.

[28] R. Szeliski, R. Zabih, D. Scharstein, O. Veksler, V. Kolmogorov, A. Agarwala, M. Tappen, and C. Rother. A Comparative Study of Energy Minimization Methods for Markov Random Fields with Smoothness-Based Priors. *TPAMI'08*.

[29] M. Tianyang and L. J. Latecki. Maximum weight cliques with mutex constraints for video object segmentation. In *CVPR'12*.

[30] R. Tokola, W. Choi, and S. Savarese. Breaking the chain: liberation from the temporal Markov assumption for tracking human poses. In *ICCV'13*.

[31] D. Weiss, B. Sapp, and B. Taskar. Sidestepping intractable inference with structured ensemble cascades. In *NIPS'10*.

[32] Y. Xu, L. Qin, and Q. Huang. Coupling Multiple Alignments and Re-ranking for Online Multi-target Tracking. In *ACCV'14*.

[33] P. Yadollahpour, D. Batra, and G. Shakhnarovich. Diverse M-best solutions in MRFs. In *NIPS Workshops'11*.

[34] B. Yang, C. Huang, and R. Nevatia. Learning affinities and dependencies for multi-target tracking using a CRF model. In *CVPR'11*.

[35] Y. Yang and D. Ramanan. Articulated pose estimation with flexible mixtures-of-parts resenting shape. In *CVPR '11*.

[36] A. Zamir, A. Dehghan, and M. Shah. GMCP-Tracker: Global Multi-Object Tracking Using Generalized Minimum Clique Graphs. In *ECCV'12*.

[37] D. Zhang, O. Javed, and M. Shah. Video Object Segmentation through Spatially Accurate and Temporally Dense Extraction of Primary Object Regions. In *CVPR'13*.

[38] L. Zhang, Y. Li, and R. Nevatia. Global data association for multi-object tracking using network flows. In *CVPR*, 2008.

[39] D. Zuckerman. Linear degree extractors and the inapproximability of max clique and chromatic number. *Proceedings of the thirty-eighth annual ACM Symposium on Theory of Computing*, 3:103–128, 2007.

[40] S. Zuffi, J. Romero, C. Schmid, and M. J. Black. Estimating Human Pose with Flowing Puppets. In *ICCV'13*.