

Multi-Target Tracking of Time-varying Spatial Patterns

Jingchen Liu¹

Yanxi Liu^{1,2}

¹Department of Computer Science and Engineering

²Department of Electrical Engineering

The Pennsylvania State University

University Park, PA 16802, USA

{jingchen, yanxi}@cse.psu.edu

Abstract

Time-varying spatial patterns are common, but few computational tools exist for discovering and tracking multiple, sometimes overlapping, spatial structures of targets. We propose a multi-target tracking framework that takes advantage of spatial patterns inside the targets even though the number, the form and the regularity of such patterns vary with time. RANSAC-based model fitting algorithms are developed to automatically recognize (or dismiss) (il)legitimate patterns. Patterns are represented using a mixture of Markov Random Fields (MRF) with constraints (local and global) and preferences encoded into pairwise potential functions. To handle pattern variations continuously, we introduce a posterior probability for each spatial pattern modeled as a Bernoulli distribution. Tracking is achieved by inferring the optimal state configurations of the targets using belief propagation on a mixture of MRFs. We have evaluated our formulation on real video data with multiple targets containing time-varying lattice patterns and/or reflection symmetry patterns. Experimental results of our proposed algorithm show superior tracking performance over existing methods.

1. Introduction

Multi-target tracking has been a standing challenge in computer vision. Its applications include surveillance, sports video and biomedical data analysis. The essence of tracking is to use available information, from both image observations and prior knowledge, to infer the target locations in time. Our prior knowledge can be encoded into various models to regulate the tracking, for example, previous work has used the adaptive appearance models [19, 11], temporal motion models (filters) [9], spatiotemporal consistency models [21], target interaction models [8, 22, 16] and global scene constraint models [1, 18]. Little attention has

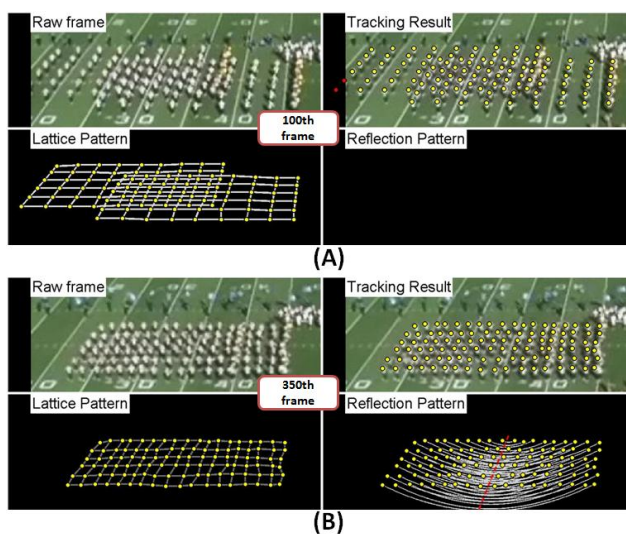


Figure 1. Tracking 108 multiple targets in a marching band video. (A): the 100th frame, two partially overlapping lattice patterns are detected and tracked; (B): the 350th frame, both a lattice pattern and a reflection symmetry pattern are found and tracked.

been paid to explore and take advantage of spatial patterns in the targets.

In this paper, we address the problem of tracking multi-targets that contain time-varying regular patterns. These are frequently seen in scenarios (Figure 2) such like moving textiles (clothes, dress, scarf), flying through city scenes, human/animal group movements (marching band in figure 1 or other types of team-movements), flying birds (geese), schooling fish and tagged cardiac MR videos.

Effective pattern recognition during multi-target tracking is novel and of utter importance for a deeper and higher level (semantic level) understanding of the multi-subjects being tracked, leading to potential social, cultural or biological implications that current tracking systems are not prepared to handle. On the other hand, multi-targets with



Figure 2. Real world examples of various time-varying lattice patterns and reflection symmetry patterns.

repeating patterns pose special computational challenges to state of the art tracking systems [10], with high ambiguous targets and dynamic topological structures.

The framework we proposed is general and the number of time-varying patterns is not limited. To start, we select two most common and mathematically well-defined types of 2D regular patterns: 2D repeating (lattice) and bilateral reflecting patterns (selected from a finite set of regular patterns) to track directly on low resolution real world video data. By defining a different set of topological constraints, future users can utilize our framework to track their own specific patterns.

To track multiple targets that have time-varying spatial patterns, we face two intertwined problems: 1) To recognize the (dis)appearance of spatial patterns from target locations; 2) To utilize pattern constraints for inferring target locations. By treating the patterns as latent variables that guide the tracking, we adopt an iterative two-step framework similar to the dynamic expectation-maximization (EM) method [7], addressing both problems simultaneously (Figure 4). In the first (E-like) step we detect possible patterns and evaluate their posterior probabilities based on previous tracking results. In the second (M-like) step, on a mixture of MRFs, we infer optimal target locations under pattern constraints controlled by posteriors as well as cues from image observations.

The idea of applying graphical models to pass messages among targets has been widely used in the multi-target tracking community. Khan et al. [8] adopt an on-the-fly MRF to model the motion prior of interacting ants; Yu and Wu [22] model target *competition* using an ad-hoc Markov network to address the problem of *coalescence* among nearby targets; Qu et al. [17] propose a Magnetic-Inertia Potential Model. All these approaches work by repelling targets that are too close to each other, however they limit their consideration to local interactions only.

For tracking topology-invariant patterns, Lin and Liu [10] propose a lattice-based MRF for tracking near-regular

textures. Park et al. [15] propose an efficient mean-shift belief propagation (MSBP) algorithm for inference on lattice-structured graphical models for multi-target tracking. Yang et al. in [20] have proposed the idea of online exploring motion correlation among several targets for more robust tracking. These algorithms do not adapt to topology or motion correlation variations during tracking.

2. Spatial Patterns

Spatially regular patterns can be formally categorized in terms of their corresponding symmetry groups. In this work, we focus on two basic types of spatial patterns: *lattice patterns* and *reflection patterns*. A perfect lattice pattern has a corresponding symmetry group with a pair of translation symmetry generators leading, naturally, to an underlying lattice structure[4] (thus the term ‘lattice pattern’); while a reflection pattern has a bilateral reflection symmetry group of order 2.

2.1. Pattern Modeling

We are interested in modeling, detecting and tracking the deformation of mathematically well-defined regular patterns in real world videos (Figure 2). Therefore, we need to represent and measure the degree of distortions of potential lattice or reflection patterns respectively. Figure 3 illustrates a snapshot of a video frame where three potential patterns encountered during tracking are detected under our proposed framework (Section 3). A key function of our approach is to monitor the birth and death of (near) regular patterns contained in the multi-target set continuously.

2.1.1 Lattice Pattern

A perfect lattice pattern (point grid) in 2D Euclidean space can be modeled by two smallest linearly independent generating vectors $T_1 = (t_{1x}, t_{1y})^T$ and $T_2 = (t_{2x}, t_{2y})^T$ [14], as well as one of the node (not unique) as its origin

$O = (o_x, o_y)^T$, so that each node v of the lattice can be indexed by a unique $(i, j) \in Z^2$

$$v(i, j) = O + iT_1 + jT_2 = (1 \ i \ j) \cdot w, \quad (1)$$

where $w = (O, T_1, T_2)^T$ is the lattice parameter and $v(i \pm 1, j)$ and $v(i, j \pm 1)$ are the T_1 -neighbors and the T_2 -neighbors of $v(i, j)$ respectively. Given the index and position of the lattice nodes $\{(i_n, j_n, v(i_n, j_n)) | n = 1, \dots, N\}$, let the generating matrix M be

$$M = \begin{bmatrix} 1 & i_1 & j_1 \\ \vdots & \vdots & \vdots \\ 1 & i_N & j_N \end{bmatrix}, \quad (2)$$

and $V = [v(i_1, j_1) \cdots v(i_N, j_N)]^T$. We can compute the minimum mean-square-error (MMSE) estimate of w as

$$\hat{w} = (M^T M)^{-1} M \cdot V, \quad (3)$$

with mean-square error distance D^2 being

$$D^2 = \frac{1}{N} \|V - M\hat{w}\|_2^2. \quad (4)$$

2.1.2 Reflection Pattern

In 2D Euclidean space, a reflection symmetry with axis l maps an arbitrary point p onto a bilateral symmetric location $p'|_l$. A reflection pattern indexed by k consists of target pairs that are symmetric w.r.t. the same axis l_k . We define a symmetry deformation distance $d_{i,j,k}$ for each pair of points (p_i, p_j) w.r.t. axis l_k as

$$d_{i,j,k} = \|p'_i|_{l_k} - p_j\|_2, \quad (5)$$

and the overall symmetry deformation distance of a whole reflection pattern G_k is the maximum deformation distance of its pairs

$$D_k = \max_{(i,j) \in E_k} d_{i,j,k}. \quad (6)$$

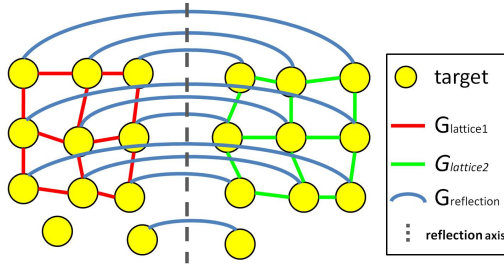


Figure 3. Sample detection result: a mixture of Markov networks containing 2 lattice patterns connected by red and green edges respectively and 1 reflection symmetry pattern connected by blue curves.

The reason for using maximum distance instead of mean squared distance is that symmetry pattern detection is likely to include some outliers. D_k defined in 6 is sensitive to the worst pair, so we can easily detect outliers and thus modify our pattern by removing them.

2.2. Pattern Recognition

Given all target locations, we apply Random Sample Consensus (RANSAC) [5] to discover patterns as subsets of all targets. Since the process of pattern (de)formation is gradual, it is not necessary to do spatial pattern recognition for each frame. Therefore we refresh the pattern set periodically and when any pattern posterior drops below some threshold. Otherwise we maintain the set of detected patterns and only evaluate the pattern posterior probabilities once the tracking result is updated.

2.2.1 Lattice Detection and Evaluation

To detect lattice structures, in each RANSAC iteration, we randomly sample one target and two of its neighbors within a search radius r to propose a unit lattice. The lattice model parameter $\{O, T_1, T_2\}$ can thus be initialized, meanwhile all other targets are assumed to be ‘outliers’. We then try to expand the lattice to its maximum along the positive and negative directions of T_1 and T_2 . During each expansion, we first predict new target locations using the current lattice model. If the nearest ‘outlier’ is within some error bound d , this target is accepted as an inlier and the lattice model is updated using equation 3. In our implementation, we use recursive least squares (RLS) [6] as an alternative of equation 3 to update the model incrementally instead of recomputing everything once new targets are accepted.

Each RANSAC iteration returns a lattice model. After enough iterations (we run 200 times in our implementation), lattices that are either too small or completely contained by bigger ones are removed. It is also forbidden for one target to belong to two lattices except on the boundary. When such conflict occurs, we keep the lattice that covers more targets.

After applying equation 3 to update pattern parameters, we use

$$p(\text{pattern}) = \exp\left\{-\frac{D^2}{2\sigma^2}\right\} \quad (7)$$

to evaluate the pattern posterior, where D is a quantitative measure of pattern deformation obtained in equation 4, and σ controls the tolerance of lattice pattern deformation.

2.2.2 Reflection Pattern Detection and Evaluation

For global reflection symmetry pattern detection, the traditional Hough transform approach [12] fails because any pair of targets (v_i, v_j) can generate a reflection axis proposal with a large portion of proposals being outliers. Therefore

we have to explore local topology to get rid of false reflection pairs in the first place. Similar to [13], in each RANSAC iteration, we randomly pick targets i and j , propose reflection axis l_{ij} and then find out the nearest neighbor of i and j , denoted by r and t respectively. If r and t are also symmetric over l_{ij} within some tolerance, we accept it as a good proposal and check all other targets to find symmetrical pairwise matches that share the same reflection axis. We acknowledge a reflection pattern if we can find a significant number of supporting pairs that share the same axis.

We reestimate the pattern parameters when the tracker states are updated. Minimizing D_k in equation 6 directly is a nonlinear problem and difficult to solve. Instead, we turn to an heuristic approach to fit a reflection axis to all mid points of the reflection pairs. The posterior probability of a reflection pattern is thus estimated by substituting equation 6 to equation 7.

3. Multi-target Tracking on a Mixture of Markov Networks

We propose a novel framework to achieve adaptive multi-pattern multi-target tracking by inferring the optimal target states on a mixture of Markov networks with posteriors estimated in the pattern detection phase.

Let $S = \{s_1, \dots, s_N\}$ and $O = \{o_1, \dots, o_N\}$ denote the states (pixel coordinates) and observations of N targets in the current frame, \hat{S}^p denote the estimated target states in the previous frame. Spatial pattern k is represented as an undirected graphical model $\mathbf{G}_k(V_k, E_k)$, where $V_k \subseteq S$ is the subset of targets that belong to pattern k . Pattern preferences and constraints are encoded into pairwise potential

functions $\psi_k(s_i, s_j)$ defined on E_k implying statistical dependency between the states of target i and j for each edge $(s_i, s_j) \in E_k$. Optimal target states can then be inferred using belief propagation along the edges.

To construct a Markov network to represent statistical dependency among the targets, we connect each target in a lattice pattern with its T_1 and T_2 neighbors. We favor multi-target configurations that appear more regular under the given lattice topology by defining

$$\psi_{lat}(s_i, s_j) = \exp\left(-\frac{\|s_i - s_j - T_m\|^2}{2\sigma_{lat}^2}\right), \quad (8)$$

where σ_{lat} controls the tolerance of the lattice deformation, and $m \in \{1, 2\}$ depending on whether target i and target j are connected via T_1 edge or T_2 edge.

Similarly, for reflection patterns, we connect each reflection symmetric pair to form a Markov network and define pairwise potential on the edges to be

$$\psi_{ref}(s_i, s_j) = \exp\left(-\frac{d_{ij,k}^2}{2\sigma_{ref}^2}\right), \quad (9)$$

with σ_{ref} controls the tolerance of the reflection deformation.

We combine all Markov networks to form a mixed network (multi-graph) $G(V, E)$, such that

$$\begin{cases} V = \bigcup_{i=1 \dots K} V_i \\ E = \{(s_i, s_j, k) | (s_i, s_j) \in E_k\} \end{cases} \quad (10)$$

We also assign pattern indicators $\gamma = \{\gamma_1, \dots, \gamma_K\}$ with $\gamma_k \in \{1, 0\}$ indicating whether detected pattern k is a true or false positive. The pattern posterior is thus a Bernoulli distribution with $p(\gamma_k)$, short for $p(\gamma_k = 1)$, representing the probability of pattern k being correct. The joint probability of target states given observations, motion prior and pattern indicators can be defined on the mixed Markov network G and factorized to

$$p(S|O, \hat{S}^p, \gamma) \propto \prod_k \prod_{(i,j) \in E_k} \psi_k^{\gamma_k}(s_i, s_j) \prod_i \phi(s_i, o_i, \hat{s}_i^p), \quad (11)$$

where $\phi(\cdot)$ is the measurement potential between a state and image observation incorporated with motion prior. Pattern indicator γ_k controls whether all edges in G_k take effect. We first apply the sum-product rule to obtain the marginal distribution of s_i

$$p(s_i|O, \hat{S}^p, \gamma) \propto \phi(s_i, o_i, \hat{s}_i^p) \prod_k \prod_{j:(j,i) \in E_k} m_{ji,k}(s_i, \gamma), \quad (12)$$

where $m_{ji,k}(s_i, \gamma)$ is the message passed from target j to target i on graph \mathbf{G}_k under pattern indicator γ_k ,

$$m_{ji,k}(s_i, \gamma) = \sum_{s_j} (\cdot)_{j \setminus i} \psi_k^{\gamma_k}(s_j, s_i), \quad (13)$$

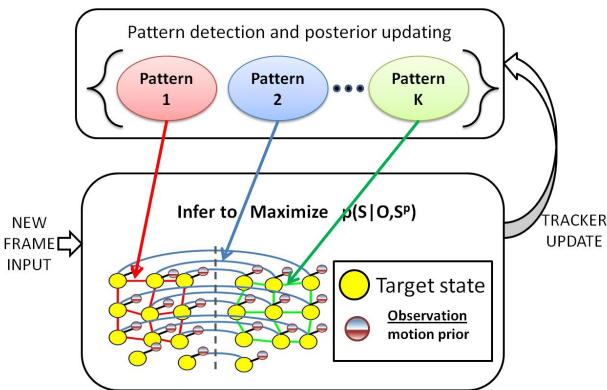


Figure 4. Our tracking framework. At E-like step, we detect patterns and/or evaluate pattern posteriors based on previous tracking results. At M-like step, we estimate the best target states given detected patterns, image observation and motion prior via inference on the mixture of Markov networks.

where we introduce the shortened notation

$$(\cdot)_{j \setminus i} = \phi(s_j, o_j, \hat{s}_j^p) \prod_{l \in N(j) \setminus i | k} m_{l,j,k_j}(s_j), \quad (14)$$

where $N(j) \setminus i | k$ represents the set of neighbors of j over all patterns, except target i over pattern k .

The sum-product rule is then used again to sum out γ

$$\begin{aligned} & p(s_i | O, \hat{S}^p) \\ = & \sum_{\gamma \in \{0,1\}^K} p(s_i | O, \hat{S}^p, \gamma) \cdot p(\gamma) \\ \propto & \sum_{\gamma \in \{0,1\}^K} \phi(s_i, o_i, \hat{s}_i^p) \prod_k \prod_{(j,i) \in E_k} m_{j,i,k}(s_i, \gamma) p(\gamma) \\ = & \phi(s_i, o_i, \hat{s}_i^p) \prod_k \prod_{(j,i) \in E_k} \sum_{\gamma_k \in \{0,1\}} m_{j,i,k}(s_i, \gamma) p(\gamma_k) \\ = & \phi(s_i, o_i, \hat{s}_i^p) \prod_k \prod_{(j,i) \in E_k} \tilde{m}_{j,i,k}(s_i, \gamma) \end{aligned} \quad (15)$$

where we have modified the message passing from target j to i by incorporating pattern posteriors into the original message

$$\begin{aligned} \tilde{m}_{j,i,k}(s_i) &= \sum_{\gamma_k \in \{0,1\}} m_{j,i,k}(s_i, \gamma) p(\gamma_k) \\ &= \sum_{s_j} (\cdot)_j \tilde{\psi}_k(s_j, s_i) \end{aligned} \quad (16)$$

where the pairwise potential on \mathbf{G}_k is also redefined by

$$\tilde{\psi}_k(s_j, s_i) = \psi_k(s_j, s_i) p(\gamma_k) + p(\bar{\gamma}_k). \quad (17)$$

It can be seen by comparing equations 15 and 16 with equations 12 and 13 that instead of doing summations over γ , which means running BP 2^K times for different configurations of γ , we can directly incorporate the pattern posterior into the pairwise potential functions via alpha-blending according to equation 17 and perform BP only once on G . Furthermore, by introducing the pattern posterior $p(\gamma)$ we make ‘soft’ decisions on whether a pattern is correct or not, and thus are able to smoothly adapt to pattern formation, deformation and reorganizations.

4. Experimental Results

4.1. Data Sets

We evaluate our approach on three archived marching band performance videos, with a total of 670, 326 and 414 frames and the number of tracked targets being 128, 192 and 108, respectively. The marching band members form time-varying lattice patterns and reflection symmetry patterns. Another test video shows parachuters forming a lattice structure, including 560 frames and 81 targets initially. All of these videos are of low resolution, with small size and similar-looking targets.

4.2. Experimental Setup

To demonstrate the contribution of modeling spatial patterns, we apply our framework on a simple tracking model by using one fixed appearance template for all targets, and by calculating the normalized cross-correlation score as the observation measurement. The state s_i of target i is its pixel coordinate in the current frame. We assume a 1st-order dynamic motion prior, modeled by $P(s_i | \hat{s}_i^p) \sim N(\hat{s}_i^p, \Sigma)$, where $N(\mu, \Sigma)$ is the normal distribution with mean μ and variance Σ . A more sophisticated tracking model with adaptive appearance updating, multi-feature fusion or temporal particle filtering could be incorporated with our framework directly.

The positions of all targets to be tracked are given in the first frame while all lattice and reflection patterns are detected automatically (Section 2). For marching band videos, we also estimate the perspective transformation of the ground plane in advance using the first frame, so that the lattice and symmetry patterns can be detected and evaluated in a rectified view.

We model the spatial interaction among targets using a mixture of three types of Markov networks: lattice pattern, reflection symmetry pattern and a close-target competition network introduced in [22], which repels targets that are too close to each other. σ_{lat} and σ_{ref} that control tolerance of pattern deformation are set to 5 and 10 respectively. When a target moves out of the image boundary, its image observation is ignored by setting observation likelihood to uniform distribution, however given the spatial pattern constraint, we can keep tracking the invisible targets.

The lattice and symmetry pattern (re)initialization is implemented using Matlab, which takes about 2 seconds for a frame containing about 100 targets. We do pattern reinitialization every 30 frames to detect newly formed patterns. The inference on a mixture of graphical models is implemented using C++, which takes less than 1 second for a frame containing about 100 targets. Considering that RANSAC-based pattern detection can be processed in parallel, and more efficient inference approaches on loopy graphical models can be adopted, e.g. [15], both parts can be further accelerated for real-time application.

4.3. Qualitative Tracking Results

Sample tracking results are shown in Figures 5 - 8. The most dominant patterns are all correctly detected, recognized and tracked. The pattern transitions are also handled ‘softly’ via the proposed inference framework on a dynamically composed mixture of Markov networks. The functionality of this framework can also be interpreted as a majority vote scheme: When a large proportion of targets in a pattern is tracked correctly, the posterior probability of the pattern is high so that the rest of the targets, when caught

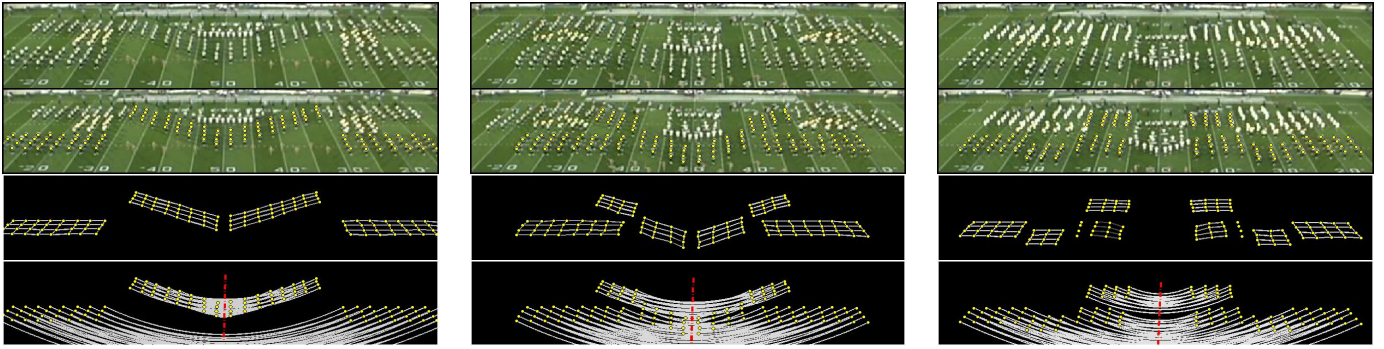


Figure 5. Detection and tracking lattice and reflection patterns on video #1. Left: 200th frame, middle: 400th frame, right: 650th frame

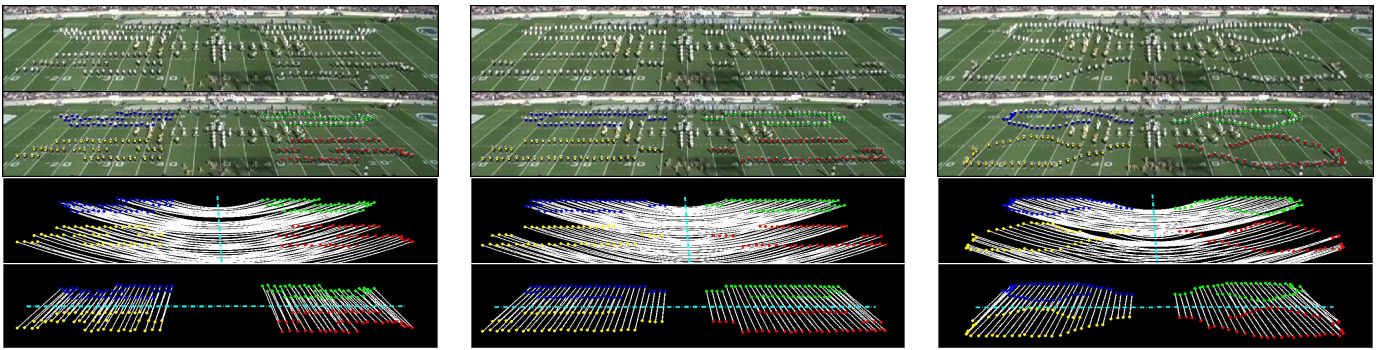


Figure 6. Detection and tracking reflection patterns on video #2. Left: 1th frame, middle: 200th frame, right: 300th frame

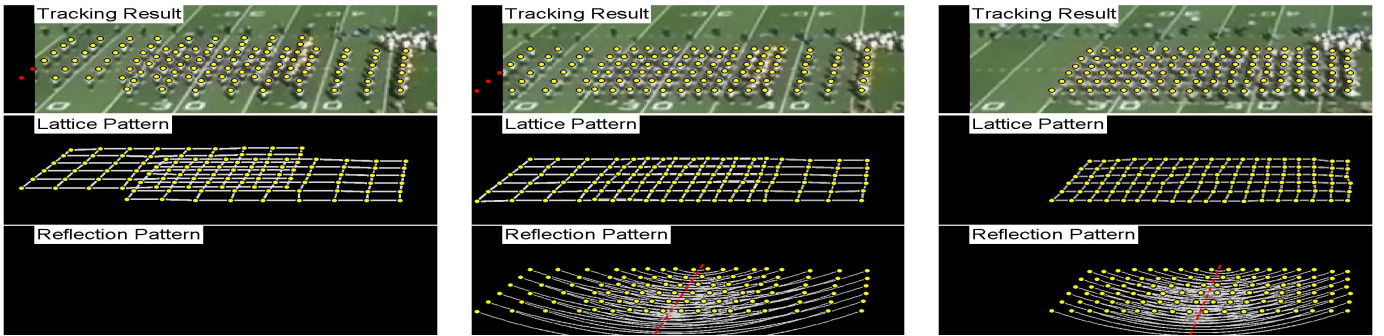


Figure 7. Detection and tracking lattice and reflection patterns on video #3. Left: 100th frame, middle: 200th frame, right: 400th frame

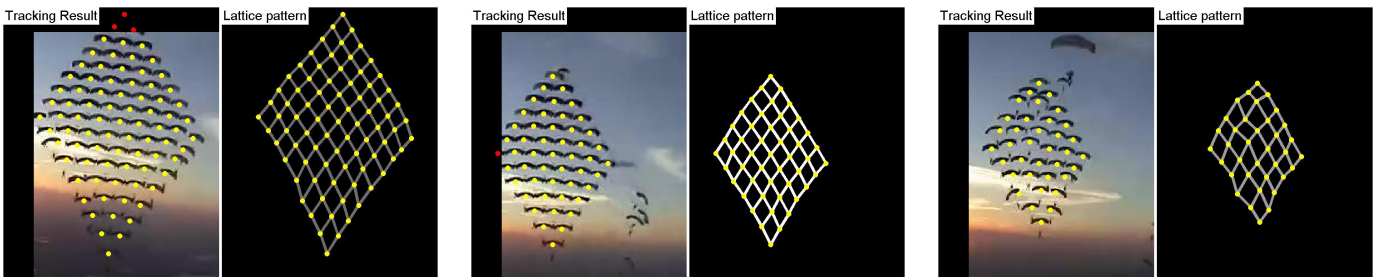


Figure 8. Tracking and monitoring a breaking away lattice pattern of a parachute video. Left: 230th frame, 81 targets; middle: 400th frame, 49 targets; right: 530th frame, only 36 targets are left

in ambiguity, can be corrected via pattern constraints. On the other hand, when the pattern is varying in time, its posterior will drop dramatically and thus the pattern constraint is decreased, leaving the individual trackers with more freedom. Figures 7 and 8 demonstrate the advantage of using spatial constraints to continue tracking targets that are out of the image boundary, and the target identities can be reestablished upon their return (Figure 8).

4.4. Quantitative Comparison

In order to make quantitative comparisons against two alternative tracking approaches, we have manually labeled the groundtruth for two of the challenging marching band videos. One alternative method applies the close-target repel model as in [22], the other applies mean-shift tracking [3] independently on each individual target with kernel bandwidth being 7 pixels.

We define a tracker as ‘lost’ if the pixel distance between the tracker’s position and the target ground truth is larger than some threshold. Comparisons of results are shown in Figures 9 and 10. A lost tracker is marked as a false positive (FP) and the corresponding ground truth location is marked as false negative (FN). In our case, the number of FP and FN are equal since the number of trackers is always equal to the number of targets. The close-repel framework works well for targets surrounded by others, who are more constrained to resolve appearance ambiguity, while targets on the boundaries are more likely to be lost due to background clutter.

We also compare with topology-fixed single lattice tracking approach (Lin & Liu PAMI07 [10]) on marching band video #3 to demonstrate the multi-lattice adaptivity of our approach to varying topologies. The result and statistics are shown in Figure 11.

Please refer to project website for related videos: <http://vision.cse.psu.edu/NRTtracking.html>

5. Conclusions & Future Work

We propose a novel inference framework to track multiple targets that form time-varying spatial patterns. Each pattern is specified with a Markov network enabling both local (short range) and global (long range) relations among multiple targets. Through inference on such a mixture of Markov networks, we effectively exploit spatial domain information and its interplay with temporal cues to achieve robust tracking and a high level of understanding of the targets tracked.

By treating spatial patterns as Bernoulli variables, we adjust the spatial pattern constraints according to their posterior probability, so that the algorithm can handle pattern transitions softly and adaptively. Experimental results on various challenging videos demonstrate the effectiveness of

our approach and superior performance of the proposed algorithm over existing methods.

Currently, our framework is applied to a sequential tracking model where the targets are updated frame by frame. We plan to incorporate detection-and-association [2] based tracking next, so that we can utilize spatiotemporal patterns directly to further enhance tracking robustness.

Acknowledgment This work was partially funded under NSF grant IIS-0729363.

References

- [1] S. Ali and M. Shah. Floor fields for tracking in high density crowd scenes. In *European Conference on Computer Vision (ECCV)*, pages 1–14, 2008. 1
- [2] M. Andriluka, S. Roth, and B. Schiele. People-tracking-by-detection and people-detection-by-tracking. In *Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, 2008. 7
- [3] D. Comaniciu, V. Ramesh, and P. Meer. Real-time tracking of non-rigid objects using mean shift. volume 2, pages 142–149, 2000. 7
- [4] H. Coxeter. *Introduction to Geometry*. Wiley, second edition, 1980. 2
- [5] M. A. Fischler and R. C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24(6):381–395, 1981. 3
- [6] M. H. Hayes. *Statistical Digital Signal Processing and Modeling*. Wiley, 1996. 3
- [7] M. Jorgensen. A dynamic EM algorithm for estimating mixture proportions. *Statistics and Computing*, 9(4):299–302, 1999. 2
- [8] Z. Khan, T. Balch, and F. Dellaert. MCMC-based particle filtering for tracking a variable number of interacting targets. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 27(11):1805–1918, 2005. 1, 2
- [9] K. Li, M. Chen, and T. Kanade. Cell population tracking and lineage construction with spatiotemporal context. In *International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, pages 295 – 302, 2007. 1
- [10] W.-C. Lin and Y. Liu. A lattice-based MRF model for dynamic near-regular texture tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 29(5):777–792, May 2007. 2, 7
- [11] Z. Lin, L. S. Davis, D. S. Doermann, and D. DeMenthon. Hierarchical part-template matching for human detection and segmentation. In *International Conference on Computer Vision (ICCV)*, pages 1–8, 2007. 1
- [12] G. Loy and J.-O. Eklundh. Detecting symmetry and symmetric constellations of features. In *European Conference on Computer Vision (ECCV)*, volume 2, pages 508–521, 2006. 3
- [13] N. J. Mitra, L. Guibas, and M. Pauly. Partial and approximate symmetry detection for 3D geometry. In *ACM Transactions on Graphics*, volume 25, pages 560–568, 2006. 4

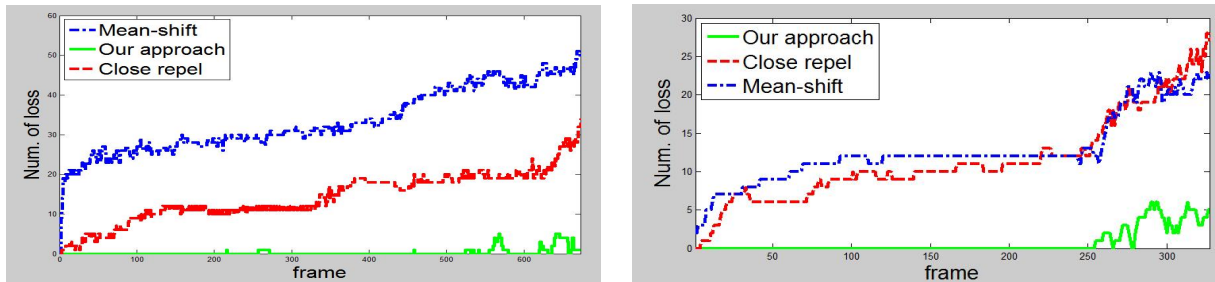


Figure 9. Comparison of number of lost targets over all frames on marching band video #1 (left) and #2 (right).

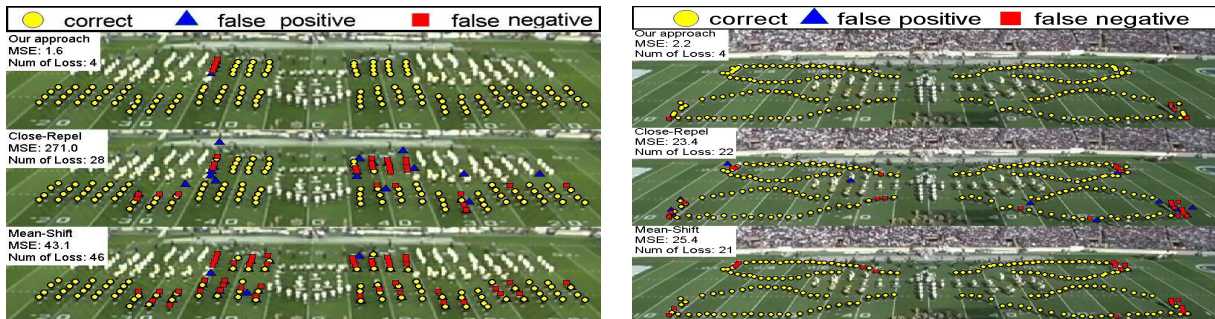


Figure 10. Comparison of tracking results on a single frame on marching band video #1 (left) and #2 (right).

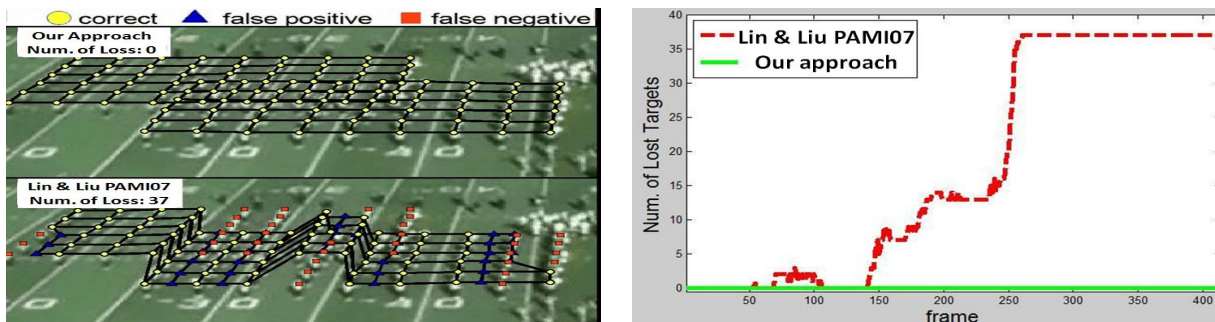


Figure 11. Comparison of tracking results (left) and statistics (right) of our approach and Lin & Liu's in PAMI07. Our approach achieves 100% accuracy during the tracking of the entire marching band video #3

[14] M. Park, K. Broeklehurst, R. Collins, and Y. Liu. Deformed lattice detection in real-world images using mean-shift belief propagation. 31(10):1804–1816, 2009. 2

[15] M. Park, Y. Liu, and R. Collins. Efficient mean shift belief propagation for vision tracking. In *Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, 2008. 2, 5

[16] S. Pellegrini, A. Ess, K. Schindler, and L. van Gool. You'll never walk alone: modeling social behavior for multi-target tracking. In *International Conference on Computer Vision (ICCV)*, 2009. 1

[17] W. Qu, D. Schonfeld, and M. Mohamed. Real-time distributed multi-object tracking using multiple interactive trackers and a magnetic-inertia potential model. *IEEE Transactions on Multimedia*, 9(3):511–519, 2007. 2

[18] M. Rodriguez, S. Ali, and T. Kanade. Tracking in unstructured crowded scenes. In *International Conference on Computer Vision (ICCV)*, 2009. 1

[19] K.-W. Wan, C.-C. Wang, and T. Ton. Weakly interacting object tracking in indoor environments. In *IEEE Workshop on Advanced robotics and Its Social Impacts (ARSO)*, pages 1–6, 2008. 1

[20] M. Yang, Y. Wu, and S. Lao. Mining auxiliary objects for tracking by multibody grouping. In *International Conference on Image Processing (ICIP)*, volume 3, pages 361–364, 2007. 2

[21] Z. Yin and R. Collins. Belief propagation in a 3D spatio-temporal MRF for moving object detection. In *Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, 2007. 1

[22] T. Yu and Y. Wu. Collaborative tracking of multiple targets. In *Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages 834–841, 2004. 1, 2, 5, 7