Structural Cues in 2D Tracking: Edge Lengths vs. Barycentric Coordinates

Nicole M. Artner[1] and Walter G. Kropatsch[1]

Vienna University of Technology
Pattern Recognition and Image Processing Group, Vienna, Austria
{artner,krw}@prip.tuwien.ac.at

Abstract. Graph models offer high representational power and useful structural cues. Unfortunately, tracking objects by matching graphs over time is in general NP-hard. Simple appearance-based trackers are able to find temporal correspondences fast and efficient, but often fail to overcome challenging situations like occlusions, distractors and noise. This paper proposes an approach, where an attributed graph is used to represent the structure of the target object and multiple, simple trackers in combination with structural cues replace the costly graph matching. Thus, the strengths of both methodologies are combined to overcome their weaknesses. Experiments based on synthetic videos are used to evaluate two possible structural cues. Results show the superiority of the cue based on barycentric coordinates and the potential of the proposed tracking approach in challenging situations.

1 Introduction

Even though there exists a vast amount of approaches for video tracking [1, 2], this field of research still has some open problems and challenges. The aim of this paper is to show, which challenges can be overcome by choosing a graph-based representation for the target object and by employing structural cues deduced from this representation in tracking. We will study the following challenges from [1]: (1) Distractors: neighboring objects with similar appearance as the target object; (2) Occlusions: varying degrees of partial occlusions; (3) Varying object pose: translation and rotation in 2D and global scaling; (4) Noise: Gaussian white noise and Salt & Pepper.

The concept of the proposed approach is to represent the target object by a graph, where its vertices represent salient features describing the target object and its edges encode their spatial relationships. Instead of graph matching, appearance-based trackers are employed to find the temporal correspondences of the vertices with the help of structural cues deduced from the graph representation. Hence, the proposed approach is able to benefit from the strengths of graph-based representations to overcome challenges during tracking (see Tab. 1). In this paper, two structural cues are compared: *edge cue* and *triangle cue*. The **contributions** of this paper are:

Table 1. Strength ⊕ and weaknesses ⊖ of simple trackers and graph-based trackers.

Simple tracker	**Graph-based tracker**
⊕ fast correspondence finding	⊖ costly graph matching
⊖ sensitive against partial occlusions	⊕ robust against partial occlusions
⊖ sensitive against noise	⊕ robust against noise
⊖ sensitive against distractors	⊕ robust against distractors

1. A novel structural cue based on barycentric coordinates (triangles);
2. Comparison of performance of two structural cues (edges and triangles);
3. Evaluation of structural cues under challenges;
4. Analysis of the influence of different parameters on the proposed method.

The edge cue is related to pictorial structures introduced in 1973 by Fischler et al., where the target object is described by a set of parts in a deformable configuration. Felzenszwalb et al. [3] continued and improved the ideas of Fischler et al. to do part-based object recognition for faces and articulated objects. Ramanan et al. apply in [4] the ideas from [3] in tracking people. In comparison to the related work, the edge cue in this paper can be calculated from arbitrary graphs and instead of using structure to verify statistical hypothesis, the proposed structural cues emerge from the underlying structure.

The triangle cue in this paper is determined from barycentric coordinates, which were introduced by August Ferdinand Möbius in 1827. Barycentric coordinates are particularly important in computer graphics, but are also used in computer vision. Salzmann et al. [5] represent surfaces as triangulated meshes and try to recover their 3D shape from 2D correspondences. Barycentric coordinates are used to describe the surface coordinates of each pixel through the triangle inside which they lie. In [6], Dornaika et al. track faces in a particle filter based framework using a statistical facial appearance model. After a general 3D face model is adapted to the face in the input video, barycentric coordinates are used to describe the position of each pixel within its associated triangle. In comparison to [5] and [6], we calculate the barycentric coordinates of vertices outside of triangles (see Fig. 1).

Overview of paper: Sec. 2 describes the proposed graph model. Sec. 3 shortly presents the appearance-based tracker. Sec. 4 introduces the edge cue and Sec. 5 the novel triangle cue. In Sec. 6 the combined iterative tracking is described. Sec. 7 covers the evaluation of the proposed structural cues. Conclusions are given in Sec. 8.

2 Structural model: attributed graph

An attributed graph $\mathbf{G}$ consists of a set of vertices $\mathbf{V}$, which are connected via a set of edges $\mathbf{E}$. The edges $\mathbf{E}$ are inserted following the rules of the*Delaunay triangulation*. Hence, there is also a set of triangles $\mathbf{F}$, where $c : \mathbf{F} \mapsto V^3; c(f) = \{v_1, v_2, v_3\}$ and $\{e_1(v_1, v_2), e_2(v_2, v_3), e_3(v_3, v_1)\} \in \mathbf{E}$ are the corresponding edges. The model stores attributes with vertices, edges and triangles.

Attributes of vertices: Each vertex $v \in \mathbf{V}$ stores a set of attributes $\{\mathbf{p}, \mathbf{B}, \mathbf{a}\}$.

$\mathbf{p} : \mathbf{V} \times \mathcal{T} \mapsto \mathbf{R}^2; \mathbf{p}(v,t) = (x,y)^T$ is the 2D position of vertex v at time $t \in \mathcal{T}$. These coordinates are updated in every iteration of the tracking algorithm.

$\mathbf{B} : \mathbf{V} \times \mathbf{F}' \mapsto \mathbf{R}^3; \mathbf{B}(v, \mathbf{F}')$ is a set of barycentric coordinates of vertex v for each triangle $f \in \mathbf{F}'$, where $\mathbf{F}' = \{f \in \mathbf{F} | v \notin c(f)\}$. The barycentric coordinates are determined during initialization and are constant over time (see Sec. 5).

$\mathbf{a} : \mathbf{V} \mapsto \mathbf{R}^n; \mathbf{a}(v)$ delivers features for vertex v from an image window $I_{n \times n}$ centered at position of $\mathbf{p}(v,0)$. It is calculated during initialization and is constant over time. Any arbitrary feature can be employed in the model.

Attribute of edges: For each edge $e = (v,w) \in \mathbf{E}$ the length $l : \mathbf{E} \times \mathcal{T} \mapsto \mathbf{R}; l(e,t) = ||\mathbf{p}(v,t) - \mathbf{p}(w,t)||_2$ is the Euclidean distance of the vertices v and w at time t. These lengths are updated at each frame to deal with global scaling.

Attribute of triangles: Each triangle $f \in \mathbf{F}$ stores the ratios of its edge lengths $\mathbf{r} : \mathbf{F} \times \mathcal{T} \mapsto \mathbf{R}^3$, where $\mathbf{r}(f,t) = \{\frac{l(e_1,t)}{l(e_2,t)}, \frac{l(e_1,t)}{l(e_3,t)}, \frac{l(e_2,t)}{l(e_3,t)}\} = \{r^t_{12}, r^t_{13}, r^t_{23}\}$ are their ratios at time t. These ratios are updated at each frame.

3 Appearance-based tracker

Mean Shift [7] is employed as appearance-based tracker to associate the vertices of the graph over time. In each frame, it finds the locally optimal position $\mathbf{p}$ for each vertex v. This is achieved in an iterative process, where starting from the position from the last frame, Mean Shift searches in a local neighborhood for a position which maximizes the similarity $\mathcal{A} : R^n \times R^n \mapsto [0,1]$ to the appearance $\mathbf{a}(v)$ of the model. Similarity in appearance is determined as follows:

$$\mathcal{A}(\mathbf{a}(v), \mathbf{I}(\mathbf{p}(v,t_i))) = 1 - \delta(\mathbf{a}(v), \mathbf{I}(\mathbf{p}(v,t_i))), \tag{1}$$

where $\mathbf{I} : \mathbf{R}^2 \mapsto \mathbf{R}^n$ extracts a feature vector around position $\mathbf{p}(v,t_i)$. δ can be any distance metric suitable for the employed features. In this paper, it is the Bhattacharyya distance as described in [7]. The offset generated at time t_i points to the position maximizing $\mathcal{A}$: $\mathbf{p}(v,t_i) = \mathbf{p}(v,t_{i-1}) + \boldsymbol{m}(v,t_i)$.

4 Structural cue based on edges

Under the assumption that the target object is rigid and its motion is limited to the image plane, the length of edges does not change over time. Fig. 1 visualizes the idea behind the edge cue. This cue has already been presented in a similar form in [8], but has been improved and simplified for this paper.

The edge cue is determined several times during the iterative process (see Sec. 6) in each frame of a video. t_i indicates a point in time within the current frame starting at time t_0. For each vertex $v \in \mathbf{V}$ an edge cue can be determined

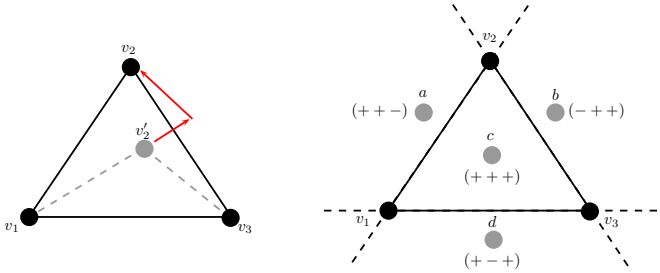

Fig. 1. Structural cues. Left: edge cue; Right: triangle cue.

based on the local, spatial deformation of graph $\mathbf{G}$. The local (deformation) energy $\mathcal{E}$ in vertex v at time t_i can be quantified as follows:

$$\mathcal{E} : \mathbf{V} \times \mathcal{T} \mapsto \mathbf{R}; \mathcal{E}(v, t_i) = \min \left(1, \sum_{e=(v,w)\in\mathbf{E}} \left| 1 - \frac{||\boldsymbol{p}(v,t_i) - \boldsymbol{p}(w,t_i)||_2}{l(e,t_0)} \right| \right). \quad (2)$$

$\mathcal{E}$ is a weight used to calculate the edge cue $\boldsymbol{d} : V \times \mathcal{T} \mapsto \mathbf{R}^2$:

$$\boldsymbol{d}(v, t_i) = \sum_{e=(v,w)\in\mathbf{E}} \mathcal{E}(w, t_i) \cdot | \, ||\boldsymbol{p}(v, t_i) - \boldsymbol{p}(w, t_i)||_2 - l(e, t_0)| \cdot \frac{\boldsymbol{p}(v,t_i) - \boldsymbol{p}(w,t_i)}{||\boldsymbol{p}(v,t_i) - \boldsymbol{p}(w,t_i)||_2}, \quad (3)$$

which is an offset vector pointing towards the structurally optimal position.

5 Structural cue based on triangles

Triangles are 2D entities, which are able to describe the geometry of planar objects and approximate curved objects (triangle mesh). By knowing the correspondence of three points at two time instances, it is possible to estimate their affine transformation in and out of the image plane.

Barycentric coordinates are an elegant way to transfer the motion information of a triangle to the neighboring vertices in a graph. The position of a vertex v can be calculated with the help of the barycentric coordinates $\{\beta_1, \beta_2, \beta_3\}$ of the three corners $c(f)$ of any triangle $f \in \mathbf{F}$. Figure 1 illustrates this concept.

During the intra-frame, iterative process, the *triangle cue* for a vertex is determined from the barycentric coordinates of a triangle $f^* \in \mathbf{F}'$. Let f^* be the triangle with the highest confidence $\mathcal{F} : \mathbf{F}' \times \mathcal{T} \mapsto \mathbf{R}$. $\mathcal{F}(f, t_i)$ is based on two properties of triangles: change of shape $\mathcal{R} : \mathbf{F} \times \mathcal{T} \mapsto \mathbf{R}$ and similarity in appearance $\mathcal{A}$ in their corners $\mathbf{c}(f)$ (see (1)). Change in ratios $\mathcal{R}$ is determined as $\mathcal{R}(f, t_i) = \min(|1 - \frac{r_{12}(t_i)}{r_{12}(t_0)}| + |1 - \frac{r_{13}(t_i)}{r_{13}(t_0)}| + |1 - \frac{r_{23}(t_i)}{r_{23}(t_0)}|, 1)$. From this, the confidence is calculated as follows:

$$\mathcal{F}(f, t_i) = \frac{1 - \mathcal{R}(f, t_i) + \min\limits_{v\in c(f)} (\mathcal{A}(\mathbf{a}(v_j), \mathbf{I}(\mathbf{p}(v_j, t_i))))}{2} \quad (4)$$

Algorithm 1 Combined, iterative tracking within one frame.

```
IterativeTracking
    ε_i, ε_A, ε_E thresholds for iterations, similarity, energy
    i ← 1                                                          ▷ iteration counter
    while i < ε_i ∧ (argmin_{v∈V}(A(a(v), I(p(v, t_i)))) < ε_A ∨ argmax_{v∈V}(E(v, t_i)) > ε_E) do
        sort V                                                     ▷ for more details see Sec. 7
        for each vertex v ∈ V do
            determine appearance cue m(v, t_i) using Mean Shift
            if i > 1 then                                          ▷ first iteration is Mean Shift only
                determine structural cue s(v, t_i)
                combine cues p(v, t_i) = p(v, t_{i−1}) + (ω · m(v, t_i) + (1 − ω) · s(v, t_i))
            else
                Mean Shift only p(v, t_i) = p(v, t_{i−1}) + m(v, t_i)
            end if
        end for
        update: A(a(v), I(p(v, t_i))), E and V of v ∈ V, F of f ∈ F
        i ← i + 1
    end while
end
update: l of e ∈ E and r of f ∈ F
```

The most stable $f^*(t_i)$ is selected by $f^*(t_i) = \underset{f \in \mathbf{F}'}{\operatorname{argmax}}(\mathcal{F}(f, t_i))$. Finally, the triangle cue $\mathbf{b} : \mathbf{F}' \times \mathcal{T} \times \mathbf{B} \mapsto \mathbf{R}^2$ is calculated from $\mathbf{B}(v, f^*)$:

$$\mathbf{b}(c(f^*), t_i, \mathbf{B}(v, f^*)) = (x, y, 1)^T = (\beta_1, \beta_2, \beta_3) \cdot \begin{pmatrix} \mathbf{p}(v_1, t_i)^T, 1 \\ \mathbf{p}(v_2, t_i)^T, 1 \\ \mathbf{p}(v_3, t_i)^T, 1 \end{pmatrix} \tag{5}$$

6 Combined, iterative tracking

The following combined, iterative tracking integrates structural cues into the mode seeking process of Mean Shift (see Sec. 3). By combining the appearance cue $\mathbf{m}(v, t_i)$ with the structural cue $\mathbf{s}(v, t_i)$ (either $\mathbf{d}(v, t_i)$ or $\mathbf{b}(v, t_i) - \mathbf{p}(v, t_{i-1})$) the proposed approach finds a position, which not only maximizes the similarity in appearance, but also the similarity in structure (shape).

During the intra-frame iterations, $\mathbf{s}$ and $\mathbf{m}$ are re-calculated and combined for each vertex v until a position $\mathbf{p}(v, t_i)$ is found where $\mathcal{E}(v, t_i) < \varepsilon_{\mathcal{E}}$ (see (2)) and $\mathcal{A}(\mathbf{a}(v), \mathbf{I}(\mathbf{p}(v, t_i))) > \varepsilon_{\mathcal{A}}$ (see (1)). $\mathbf{p}(v, t_i) = \mathbf{p}(v, t_{i-1}) + (\omega \cdot \boldsymbol{m}(v, t_i) + (1 - \omega) \cdot \boldsymbol{s}(v, t_i))$, where ω is a weight defining the influence of appearance $\mathbf{m}$ and structure $\mathbf{s}$ on the new position.

There are three ways to come up with ω: (i) similarity in appearance $\mathcal{A}$ (see 1), (ii) energy in a vertex $(1 - \mathcal{E})$ (see 2) or (iii) confidence in a vertex $\mathcal{V}$ (see 6). The confidence $\mathcal{V}$ of vertex is determined by combining similarity and energy:

$$\mathcal{V} : \mathbf{V} \times \mathcal{T} \mapsto \mathbf{R}; \mathcal{V}(v, t_i) = \frac{\mathcal{A}(\mathbf{a}(v), \mathbf{I}(\mathbf{p}(v, t_i))) + (1 - \mathcal{E}(v, t_i))}{2}, \tag{6}$$

In Alg. 1, there are two categories of updates: intra-frame and inter-frame. The **intra-frame updates** on $\mathcal{A}$, $\mathcal{E}$, $\mathcal{V}$ and $\mathcal{F}$ are necessary for the combined, iterative process, and the **inter-frame updates** on the lengths l and ratios $\mathbf{r}$ are necessary to adjust the model to global scaling.

Regular-sized triangulation	Irregular-size triangulation
$\mathbf{p}(v_1,0) = (10,50)^T$ $\mathbf{p}(v_2,0) = (40,50)^T$ $\mathbf{p}(v_3,0) = (70,50)^T$ $\mathbf{p}(v_4,0) = (40,10)^T$ $\mathbf{p}(v_5,0) = (40,90)^T$ $\mathbf{p}(v_6,0) = (25,30)^T$ $\mathbf{p}(v_7,0) = (55,30)^T$ $\mathbf{p}(v_8,0) = (25,70)^T$ $\mathbf{p}(v_9,0) = (55,70)^T$	$\mathbf{p}(v_1,0) = (60,20)^T$ $\mathbf{p}(v_2,0) = (30,80)^T$ $\mathbf{p}(v_3,0) = (100,90)^T$ $\mathbf{p}(v_4,0) = (65,100)^T$ $\mathbf{p}(v_5,0) = (10,125)^T$ $\mathbf{p}(v_6,0) = (105,45)^T$ $\mathbf{p}(v_7,0) = (130,75)^T$ $\mathbf{p}(v_8,0) = (130,110)^T$ $\mathbf{p}(v_9,0) = (10,80)^T$

Fig. 2. Graphs of synthetic sequences with their vertices at time $t = 0$. Please note that the proposed approaches is not limited to graphs with 9 vertices.

Table 2. Videos used in evaluation are made up of every possible combination in this table. T = Translation; R = Rotation; S = Scaling; D = Distractors; O = Occlusion

Layout of G	**2D Transformations in each frame**	**Challenges**
regular-sized	$T = (5,4)^T$	D
9 vertices		D; O: 1 vertex
	$T = (7,5)^T$; $R = \begin{pmatrix} \cos(10^\circ) & \sin(10^\circ) \\ -\sin(10^\circ) & \cos(10^\circ) \end{pmatrix}$	D; O: 3 vertices
		D; O: 6 vertices
irregular-sized		
9 vertices	$T = (2,1)^T$; $R = \begin{pmatrix} \cos(5^\circ) & \sin(5^\circ) \\ -\sin(5^\circ) & \cos(5^\circ) \end{pmatrix}$;	D; Gaussian white noise
	$S = \begin{pmatrix} 1.02 & 0 \\ 0 & 1.02 \end{pmatrix}$	D; Salt & Pepper 10 %

7 Evaluation of structural cues

Tab. 2 shows information about the 36 synthetic videos (size = 400×600) which are used for this evaluation. Fig. 2 visualizes the two graphs used in the synthetic videos. All vertices have the same appearance $\mathbf{a}(v)$, which makes it difficult for trackers to distinguish between them (challenge: distractors). As a feature, we extracted weighted color histograms around the position of each vertex in a 11×11 neighborhood. Three different choices for ω are evaluated: $0 = \mathcal{A}$; $1 = (1 - \mathcal{E})$; $2 = \mathcal{V}$. Additionally, three different orderings of $v \in \mathbf{V}$ are studied: 0 = fixed ordering; 1 = ascending by $\mathcal{V}$; 2 = descending by $\mathcal{V}$. This results in nine different sets of parameters $\{00, 01, 02, 10, 11, 12, 20, 21, 22\}$ and 324 $(36 \cdot 9)$ test cases for each cue.

The results can be seen in Fig. 3 and 4, where the curves visualize the mean error (Euclidean distance from ground truth position averaged over all vertices in graph) in a vertex at each frame. For both structural cues, the choice of ω and the ordering of $\mathbf{V}$ have a noticeable influence on the results. For all test cases, the best result of the triangle cue is superior against the best result of the edge cue. The best parameter set for the triangle cue is $\{20\}$ and the worst is $\{00\}$. For the edge cue the best is $\{00\}$ and the worst $\{10\}$. The best parameters for the triangle cue are able to achieve a total error (summed over all test cases) of only ≈ 345, whereas the best edge cue results in ≈ 1994.

There are several drawbacks to the edge cue: The quality of this structural cue highly depends on the layout of the edges in the graph. Furthermore, as edges are a one dimensional entity, they are only capable of providing distance information. As this cue is local, there is no direct influence from vertices further

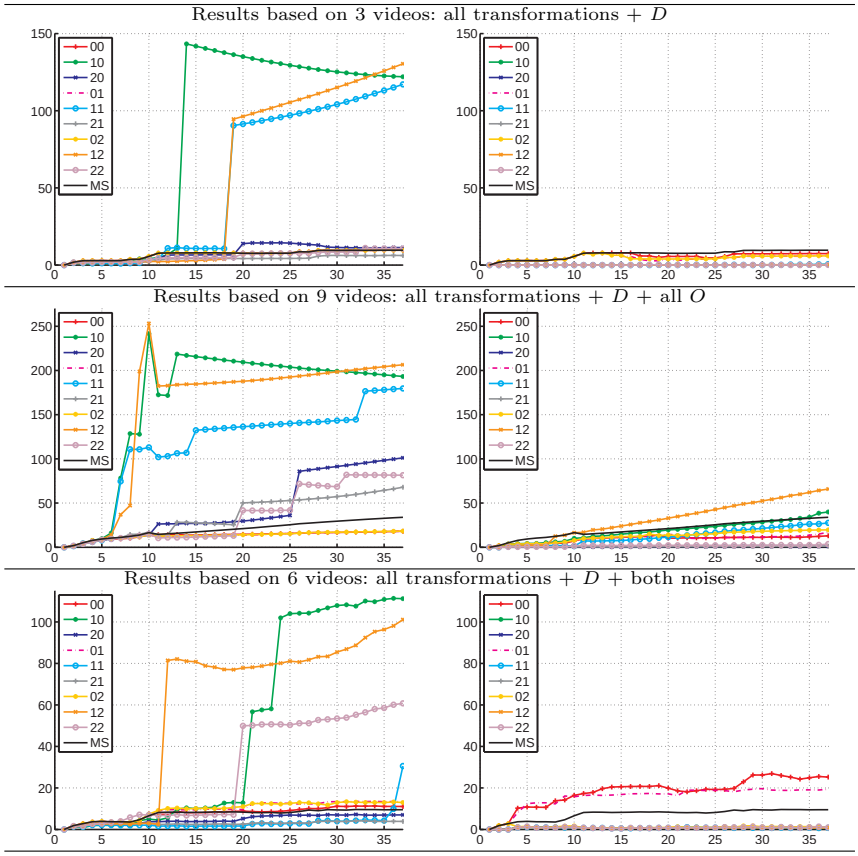

Fig. 3. Results with regular-sized triangulation. Left: edge cue; Right: triangle cue. Vertical axis: error; Horizontal axis: frame. MS = Mean Shift (baseline approach).

away in the graph. Information propagates throughout the whole graph, but in challenging cases this can be problematic.

8 Conclusions and future work

In this paper, we studied the potential of structural cues in 2D tracking of multiple targets. An attributed graph acted as a model describing the structure of the target object. Iterative tracking combined hypotheses of the appearance-based trackers with the structural cues deduced from the model to establish temporal correspondences. This paper evaluated two different structural cues: edge cue and triangle cue. The results of the evaluation showed the superiority of the triangle cue. In the future, we plan to apply the triangle cue in tracking articulated objects and extend this approach to 3D.

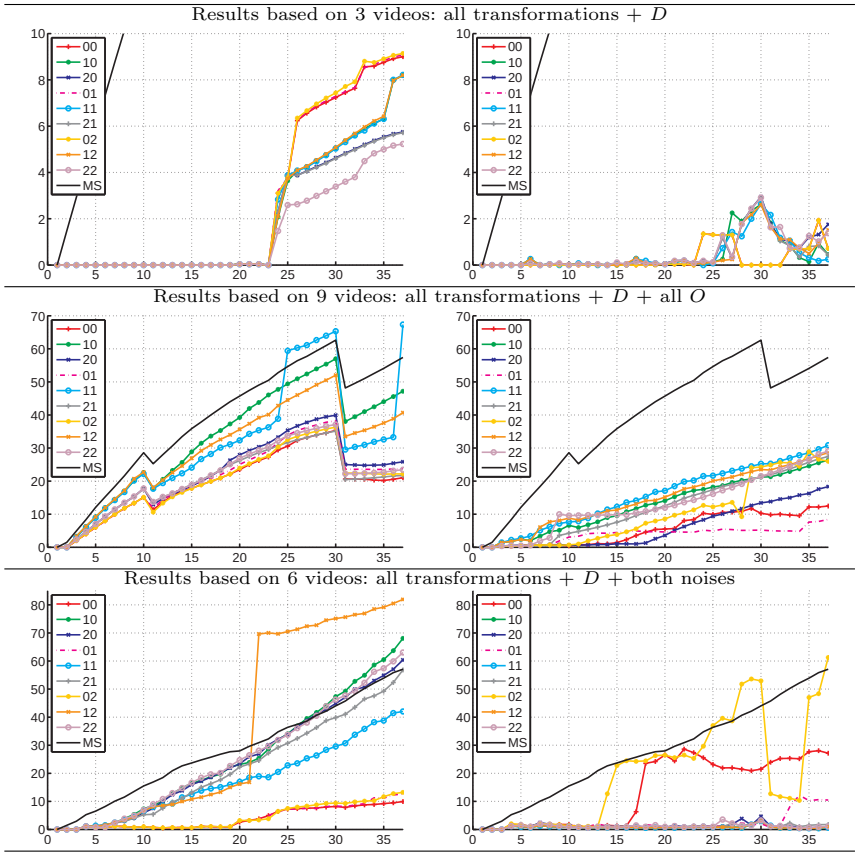

Fig. 4. Results with irregular-sized triangulation. Left: edge cue; Right: triangle cue. Vertical axis: error; Horizontal axis: frame. MS = Mean Shift (baseline approach).

References

1. Maggio, E., Cavallaro, A.: Video Tracking: Theory and Practice. Wiley (2011)
2. Yilmaz, A., Javed, O., Shah, M.: Object tracking: A survey. ACM Comput. Surv. **38**(4) (2006)
3. Felzenszwalb, P.F.: Pictorial structures for object recognition. IJCV **61** (2005) 55–79
4. Ramanan, D., Forsyth, D.: Finding and tracking people from the bottom up. In: CVPR. Volume 2., IEEE (2003) 467–474
5. Salzmann, M., Hartley, R., Fua, P.: Convex optimization for deformable surface 3-d tracking. In: 11th International Conference on Computer Vision, IEEE (2007) 1–8
6. Dornaika, F., Davoine, F.: On appearance based face and facial action tracking. Circuits and Systems for Video Technology **16**(9) (2006) 1107–1124
7. Comaniciu, D., Ramesh, V., Meer, P.: Kernel-based object tracking. PAMI **25**(5) (2003) 564–575
8. Artner, N.M., Ion, A., Kropatsch, W.G.: Multi-scale 2d tracking of articulated objects using hierarchical spring systems. PR **44**(4) (2011) 800–810