**Int J Comput Vis**
DOI 10.1007/s11263-014-0740-6

# Locally Orderless Tracking

**Shaul Oron · Aharon Bar-Hillel · Dan Levi · Shai Avidan**

Received: 30 April 2012 / Accepted: 13 June 2014

**Abstract** Locally Orderless Tracking (LOT) is a visual tracking algorithm that automatically estimates the amount of local (dis)order in the target. This lets the tracker specialize in both rigid and deformable objects on-line and with no prior assumptions. We provide a probabilistic model of the target variations over time. We then rigorously show that this model is a special case of the Earth Mover's Distance (EMD) optimization problem where the ground distance is governed by some underlying noise model. This noise model has several parameters that control the cost of moving pixels and changing their color. We develop two such noise models and demonstrate how their parameters can be estimated on-line during tracking to account for the amount of local (dis)order in the target. We also discuss the significance of this on-line parameter update and demonstrate its contribution to the performance. Finally we show LOT's tracking capabilities on challenging video sequences, both commonly used and new, displaying performance comparable to state-of-the-art methods.

**Keywords** Tracking · EMD

S.Oron
Tel Aviv University
Tel Aviv 69978, Israel
E-mail: shauloro@post.tau.ac.il

A.Bar-Hillel
Microsoft Research, Advanced Technology Labs Israel Microsoft
Haifa RD Center, Building No. 23, Matam, Haifa 31905, Israel
E-mail: aharonb@microsoft.com

D.Levi
General Motors Advanced Technical Center
Hamada 7, Herzliya, Israel
E-mail: dan.levi@gm.com

S.Avidan
Tel Aviv University
Tel Aviv 69978, Israel
E-mail: avidan@eng.tau.ac.il

## 1 Introduction

When addressing the visual tracking problem one often makes an explicit or implicit assumption about the type of target being tracked, treating it as either a rigid object or a deformable one. For example, when tracking a rigid object, where the only change in appearance is due to rigid geometric transformations, it is reasonable to use a method such as template matching where the location of pixels is fixed and governed by a geometric transformation and similarity is reduced to per-pixel intensity difference. If, on the other hand, the object is extremely deformable, then tracking based on color histogram matching might be more suitable reducing the similarity between target and candidate to similarity between their color distributions.

In this work we present a novel visual tracking algorithm we call Locally Orderless Tracking (LOT). This algorithm uses a joint spatial-appearance space representation and is able to estimate, on-line, the amount of local (dis)order in the target. Thus if the target is rigid and there is little or no local disorder then LOT preserves spatial information like template matching. However, if the target is nonrigid, LOT disregards spatial information as in histogram matching.

The first contribution of our work is a new probabilistic interpretation of the Earth Mover's Distance (EMD) that we name Locally Orderless Matching (LOM). Using this interpretation one can calculate the likelihood of patch $P$ being a noisy replica of patch $Q$ where noise can be introduced by change in the spatial order of pixels in the patch, change in their appearance, or both. In other words, LOM infers the probability $Pr(P|Q, \Theta)$ where $\Theta$ are noise model parameters, some of which control the cost of moving pixels spatially while others control the cost of changing a pixels appearance, for example due to illumination variation. Since our derivation is general one can plug in any noise model into this framework and we demonstrate the use of two such noise models.

The second contribution of our work is introducing Locally Orderless Tracking which applies Locally Orderless Matching to visual tracking. Locally Orderless Tracking is a particle filter based tracker that uses Locally Orderless Matching to infer the likelihood of each observed particle being a noisy replica of the target. Particles are represented as signatures in a joint spatial-appearance space, using superpixels for better efficiency. Key to our approach is the ability to adapt to both rigid and deformable targets. This ability is obtained by an on-line noise model parameter estimation scheme driven by the LOM solution. This adaptation is a fully automated process that requires no user intervention.

This work is an extension of Oron et al. (2012), providing additional experiments and more discussions.

The rest of this paper is organized as follows. Section 2 covers related work. Section 3 presents Locally Orderless Matching. Section 4 discusses noise models. Section 5 introduces Locally Orderless Tracking. Section 6 covers experiments and we conclude in section 7.

## 2 Related Work

We are inspired by the work of Koenderink and Van Doorn (1999) on the structure of locally orderless images which proposes an image representation method where the amount of spatial order preserved globally and locally can be tuned using two parameters. This representation was shown by Ginneken and Haar Romeny (1999) to be useful for applications such as adaptive histogram equalization, noise removal and segmentation. In our case, we wish to determine the optimal extent of local disorder of the data for the purpose of tracking.

In rigid object tracking one usually attempts to exploit spatial information in the object by using template based methods. In some cases the template is used in a simple manner (Hager and Belhumeur (1998)) while others use multiple templates and sparse representations (Ross et al. (2004, 2007); Mei et al. (2011); Kwon and Lee (2010)). These approaches offer good stability and can handle occlusions and scale estimation but are less suitable for handling non-rigid deformations and dynamics such as out-of-plane-rotations.

When tracking deformable objects one often uses histogram representations (Comaniciu (2002)) or discriminative methods that treat the problem as a pixel-wise binary classification problem (Godec et al. (2011); Avidan (2005); Grabner et al. (2006)). These approaches mostly disregard spatial order, and can therefore handle difficult non-rigid transformations. However they are more prone to drift and are often less stable especially at scale estimation or occlusion handling.

Some attempt to combine rigid and deformable object approaches. For example Wang et al. (2011) use mid level cues that capture spatial information to some extent while Santner et al. (2010) heuristically combine discriminative and generative components . However, unlike LOT, these methods do not measure nor adapt to local disorder in the data in an explicit manner.

The work most related to ours is that of Elgammal et al. (2003), proposing a tracker that uses a joint spatial-appearance space and can specialize to either histogram tracking or sum-of-square-difference (SSD) tracking by an *off-line* adjustment of parameters. The proposed method is significantly different in several ways. First and foremost, due to the on-line parameter estimation which enables LOT to specialize in rigid template tracking or deformable object tracking on-line and secondly due to the use of Particle Filtering and EMD instead of the kernel based gradient decent approach of Elgammal et al.

The Earth Mover's Distance (EMD) has a long history in computer vision. EMD was first considered by Peleg et al. (1989) as an image similarity metric and popularized by Rubner et al. (2000) (who coined the name) for content based image retrieval. A probabilistic analysis of EMD and its relation with the Mallows distance was proposed by Levina and Bickel (2001) although that analysis differs from the proposed probabilistic framework which introduces a noise process that governs the ground distance in the EMD. Recently, Zhao et al. (2010) proposed a differential EMD approach that derives a gradient descent method to find the object location quickly using the EMD as a similarity measure. However, the focus of that paper is on using EMD to handle illumination changes, the object is represented as a color signature and no consideration is given to pixels inner location in the template.

Generative probabilistic Bayesian approaches also known as Particle Filters or Sequential Monte Carlo (Doucet et al. (2001)) are widely used for visual tracking (Kwon and Lee (2010); Ross et al. (2007)). In our work we closely follow the Condensation algorithm proposed by Isard and Blake (1998) which suggest a Particle Filtering technique using factored sampling.

Superpixels first proposed by Ren and Malik (2003) have been used in recent years for many computer vision applications such as segmentation and classification (Hoiem et al. (2005); He et al. (2006)) and tracking (Wang et al. (2011)). In our work, similar to Boltz et al. (2010), superpixels are used to reduce the computational cost of EMD.

We refer interested readers to a thorough survey of the vast work in visual tracking done by Yilmaz et al. (2006).

## 3 Locally Orderless Matching

Locally Orderless Matching measures the similarity between two images or two image patches based on the EMD. Pixels are represented in a joint spatial-appearance domain. For appearance we use color values but other descriptors such as gradients or local texture can also be used. For position pixel coordinates in a patch, normalized to the range $[0,1]$, are taken. A pixel is represented as $p_i = (p_i^L, p_i^A)$ where $p_i^L = (x,y)$ is the pixels location and $p_i^A \in \mathbb{R}^D$ its appearance.

We want to probabilistically explain a candidate patch $P$ as a noisy replica of the template $Q$. We begin by looking at the pixel-wise inference problem, where patches $P$ and $Q$ are treated as sets of pixels, and show that in this case the problem is equivalent to a form of EMD optimization problem. We then propose using signature representations for $P$ and $Q$ in which pixels are clustered together using superpixel segmentation and claim the problem can now be formulated as the signature EMD problem (Rubner et al. (2000)). This is done in order to reduce the computational cost of EMD and we justify it by bounding the error resulting from the related coarsening of the representation.

Let us consider patches $P$ and $Q$ as sets of pixels. We start with a probabilistic perspective of EMD and wish to show that it measures the conditional probability of one set, given the other set and model parameters. Formally, denote the two sets by $P = \{p_i\}_{i=1}^n, Q = \{q_i\}_{i=1}^n$, and assume that we have a probabilistic model stating the probability that a specific element $p \in P$ originated from a specific element $q \in Q$, $Pr(p|q,\Theta)$, with $\Theta$ the model parameters. We want to extend it to the conditional probability between the sets $Pr(P|Q,\Theta)$.

The extension relies on a hidden 1:1 mapping between elements of $P$ and $Q$. Denote such a mapping by $h : \{1,..,n\} \rightarrow \{1,..,n\}$ with $h(i) = j$ meaning that element $p_i$ was generated from element $q_j$. We can get the probability of $P$ being generated from $Q$ by marginalizing over the possible hidden assignments (dropping $\Theta$ from the notation as it is currently constant):

$$Pr(P|Q) = \sum_h Pr(P|Q,h)Pr(h) \tag{1}$$

Assuming a uniform prior over the $h$'s (no reason to assume anything else) we have:

$$Pr(P|Q) = \frac{1}{n!}\sum_h Pr(P|Q,h) \tag{2}$$

Approximating the average using maximum a posteriori (MAP) estimation, i.e. assuming the sum is dominated by the highest term (the best hidden map) we get:

$$Pr(P|Q) \sim c \cdot \max_h Pr(P|Q,h) \tag{3}$$

Dropping the constant $c$, assuming independence between the set elements and taking the logarithm we get:

$$\begin{aligned} logPr(P|Q) &\sim \max_h logPr(P|Q,h) \\ &= \max_h \textstyle\sum_{i=1}^n logPr(p_i|q_{h(i)},\Theta) \end{aligned} \tag{4}$$

**Proposition 1** *Optimization problem* (4) *is the signature EMD problem EMD(P,Q,d) for the following signatures and ground distance:*

$$\begin{aligned} P &= \{(p_1,1),(p_2,1),\ldots,(p_n,1)\} \\ Q &= \{(q_1,1),(q_2,1),\ldots,(q_n,1)\} \\ d(p,q) &= -logPr(p|q,\Theta) \end{aligned} \tag{5}$$

*Where the signatures are comprised of objects, e.g.* $(p_i, w_i)$*, each having a description* $p_i$ *and weight* $w_i$*. In our case the signatures are simply collections of all the pixels in patches* $P$ *and* $Q$ *equally weighted.*

*Proof* Starting with Equation (4) we have:

$$\begin{aligned} &\max_h \sum_{i=1}^{n} logPr(p_i|q_{h(i)},\Theta) = \\ &\min_h \sum_{i=1}^{n} -logPr(p_i|q_{h(i)},\Theta) = \\ &\min_h \sum_{i=1}^{n} d(p_i,q_{h(i)}) \end{aligned} \tag{6}$$

where the mapping $h$ can be expressed as a permutation matrix $F$ in which $f_{ij} = 1$ iff $h(i) = j$. Denoting $d_{ij} = d(p_i, q_j)$ the problem statement becomes:

$$\begin{aligned} &\min \textstyle\sum_{i,j} f_{ij} d_{ij} \\ \text{such that} \\ &\textstyle\sum_i f_{ij} = 1, \sum_j f_{ij} = 1, f_{ij} \in \{0,1\} \end{aligned} \tag{7}$$

If we put this integer linear programming problem in the canonical form $\{\min c \cdot x | Ax = b, x \geq 0\}$ we find that the matrix $A$ is totally unimodular (Heller and Tompkins (1956)). This implies the linear programming problem in which we relax the constraint $f_{ij} \in \{0,1\}$ to $f_{ij} \geq 0$ has an integral optimum, meaning the constraint can be relaxed without changing the result.

The linear programming problem obtained by this relaxation is identical to the one obtained for signature EMD with identical mass as presented by Rubner et al. (2000).

$$\begin{aligned} &\min \sum_{i,j} f_{ij} d_{ij} \\ \text{such that} \\ &f_{ij} \geq 0, \textstyle\sum_i f_{ij} \leq w_{q_j}, \sum_j f_{ij} \leq w_{p_i} \\ &\sum_{i,j} f_{ij} = \min(\textstyle\sum_i w_{p_i}, \sum_j w_{q_j}) \end{aligned} \tag{8}$$

Where in our case all $w_{p_i}$ and $w_{q_j}$ are equal to 1. In which case the inequalities $\sum_i f_{ij} \leq w_{q_j}, \sum_j f_{ij} \leq w_{p_i}$ can be replaced by equalities and then the last constraint can be dropped. □

In other words, conditional set probability, under 1:1 mapping and element independence assumptions, is equivalent to signature EMD with singleton bins. However, the equivalence naturally extends to conditional probabilities with $P$ and $Q$ containing repeating elements and signature EMD with general integer bin quantities.

**Proposition 2** *Let*

$$\begin{aligned} P &= \{(p_1, w_1^p),(p_2, w_2^p),\ldots,(p_{n_1}, w_{n_1}^p)\} \\ Q &= \{(q_1, w_1^q),(q_2, w_2^q),\ldots,(q_{n_2}, w_{n_2}^q)\} \end{aligned} \tag{9}$$

*be signatures for which we cluster repeating elements into single objects increasing their weights accordingly (i.e. $p_1$ appears $w_1^p \in \mathbb{N}$ times in P, etc.). Solving the pixel matching problem for P and Q as formulated in optimization problem* (7) *(which has $m^2$ variables where $m = \sum_{i=1}^{n} w_i^p$) is equivalent to solving the EMD problem* (8) *for P and Q (which has $n_1 \cdot n_2$ variables) i.e. both problems have the same minima.*

The proof of proposition 2 is given in the appendix. We see that when sets $P$ and $Q$ contain identical items it lowers the computational cost of the matching using EMD formulation. Hence clustering similar items and replacing them with a single object is an attractive approximation to the likelihood. However this approximation degrades as the clustering becomes coarser. We can bound this error in likelihood estimation as follows:

**Proposition 3** *Assuming that the ground distance $d(p, q)$ is a metric.*
*Let $P = \{(p_1, w_1^p), \ldots, (p_{n_1}, w_{n_1}^p)\}$ , $Q = \{(q_1, w_1^q), \ldots, (q_{n_2}, w_{n_2}^q)\}$ be two signatures and let $\widehat{P}, \widehat{Q}$ be crude versions of $P, Q$ such that any object in $\widehat{P}$ is created by uniting objects in $P$ and the same holds for $\widehat{Q}, Q$. Denote by $h_p, h_q$ the functions mapping each object $P, Q$ to its containing object in $\widehat{P}, \widehat{Q}$. Then:*

$$\begin{array}{l} |EMD(P, Q, d) - EMD(\widehat{P}, \widehat{Q}, d)| \leq \\ \sum_{i=1}^{n_1} w_i^p d(p_i, \widehat{p}_{h_p(i)}) + \sum_{i=1}^{n_2} w_i^q d(q_i, \widehat{q}_{h_q(i)}) \end{array} \tag{10}$$

In other words, the EMD approximation gap is bounded by the sum of distances between the original cluster centers and their cruder counterparts in the crude signatures. The proof is given in the appendix.

## 4 Noise Model

We have shown that Locally Orderless Matching attempts to explain a set $P$ as a noisy replica of set $Q$, under some pixel-pair noise model with parameters $\Theta$. We now present and discuss the Gaussian noise model for pixel-pairs. We give special attention to the parameter estimation scheme, demonstrating how these noise model parameters and their estimation allow our algorithm to adapt, on-line, to both rigid and deformable objects.

We note that since the derivation presented in section 3 is general with respect to the noise model, any distribution can be used as a noise model. One can use prior knowledge, theoretical or empirical, about the noise to make an educated choice. Furthermore, since our problem is cast in term of probabilistic inference the noise model parameters can be inferred using maximum-likelihood (ML) estimation based on the EMD solution, as we demonstrate for the models we present here.

We focus on the Gaussian noise model as it is simple and intuitive and also as it was found to be empirically superior to a second noise model we tested, as presented in section 6.5. Information on additional noise models is provided in appendix A.

### 4.1 Gaussian Noise

A Gaussian distribution with zero mean and scalar covariance is considered for both location and appearance, assuming independence between the two, i.e. $Pr(p|q, \Theta_L, \Theta_A) = Pr(p^L|q^L, \Theta_L) \cdot Pr(p^A|q^A, \Theta_A)$, in which case we have:

$$\begin{array}{l} Pr(p^L|q^L) \sim N(0, \Sigma_L = \sigma_L \cdot I) \\ Pr(p^A|q^A) \sim N(0, \Sigma_A = \sigma_A \cdot I) \end{array} \tag{11}$$

Denoting $\Theta = (\sigma_L, \sigma_A)$. The conditional probability is:

$$Pr(p|q,\Theta) = \frac{1}{2\pi\sigma_L^2} e^{-\frac{||p^L - q^L||_2^2}{2\sigma_L^2}} \cdot \frac{1}{(2\pi)^{D/2}\sigma_A^D} e^{-\frac{||p^A - q^A||_2^2}{2\sigma_A^2}} \tag{12}$$

Ground distance in this case is:

$$d(p,q) = \frac{1}{2\sigma_L^2}||p^L - q^L||_2^2 + \frac{1}{2\sigma_A^2}||p^A - q^A||_2^2 + C \tag{13}$$

Where $C = \frac{D+2}{2}log(2\pi)+2log(\sigma_L)+Dlog(\sigma_A)$. This model is simple and intuitive, closely related to Koenderink and Van Doorn (1999) locally orderless image representation.

Observing equation (13) it is easy to see that if $\sigma_A >> \sigma_L$ the ground distance is dominated by the first term, i.e. the cost of moving a pixels spatially is much higher than the cost of appearance errors. In this case the optimal EMD solution would leave all the pixels in place, reducing to a sum-of-square-difference, as in rigid template matching. If however $\sigma_L >> \sigma_A$ then the ground distance is dominated by the second term, making it very costly to change pixels appearance compared with moving them spatially. In this case all spatial information is lost and the EMD is reduced to histogram matching (in the EMD sense).

This property is key to LOT's ability to adapt to both rigid and deformable targets. Moreover, since the noise model parameter update is done on-line, based on the EMD solution obtained at each frame, the "rigidity" adaptation is also done on-line in a fully automatic manner requiring no user intervention and very little parameter tuning.

### *4.1.1 Gaussian Noise Parameter Estimation*

Locally Orderless Matching with a Gaussian noise model of the form discussed above has two parameters $\sigma_A$ and $\sigma_L$. Due to the independence assumed between appearance and location each parameter can be estimated separately using the same Maximum Likelihood (ML) estimator. Therefore, $p,q,\sigma,D$ will be used without the superscripts $A,L$. Recall from propositions 1,2 that $\log Pr(P|Q,\Theta) \sim \sum\limits_{i,j} d_{ij} f_{ij}$, where the $f_{ij}$ providing the mapping, are obtained from the EMD solution. Maximum likelihood can hence be obtained by differentiating $\sum\limits_{i,j} d_{ij} f_{ij}$ with respect to $\sigma$ and comparing to zero. For $d_{ij} = d(p_i, q_j) = \frac{1}{2\sigma^2}||p_i - q_j||_2^2 + \frac{D}{2}log(2\pi) + Dlog(\sigma)$ we get:

$$\sigma^2 = \frac{1}{D} \frac{\sum\limits_{i,j} f_{ij} ||p_i - q_j||_2^2}{\sum\limits_{i,j} f_{ij}}. \tag{14}$$

## 5 Locally Orderless Tracking

Locally Orderless Tracking applies Locally Orderless Matching to tracking. This is done in a Baysian approach using Particle-Filtering (PF) where the likelihood that a certain particle has originated from the tracked object is inferred using Locally Orderless Matching. The overall algorithm is given in Algorithm 1. Specific details are provided below.

**Algorithm 1** ***Locally Orderless Tracking***

```
Input: Frame $I^{(n)}$, target signature $Q_0 = \{q_i, w_i^q\}_{i=1}^{M_{Q_0}}$, noise parameters $\Theta^{(n-1)}$, particle states $\{X_k^{(n)}\}_{k=1}^N$
Output: New target state $X_{Target}^{(n)}$, updated parameters $\Theta^{(n)}$, new particle states $\{X_k^{(n+1)}\}_{k=1}^N$

1. Partition ROI in $I^{(n)}$ into superpixels $I_{SP}$
2. For each particle $X_k^{(n)}$ do:
   (a) Build signature $P_k^{(n)} = \{p_i^k, w_i^{p^k}\}_{i=1}^{M_{P_k}}$ using $I_{SP}$
   (b) Compute ground distances:
       $\{d_k\}_{ij} = d(p_i^k, q_j) = -log(p_i^k|q_j, \Theta^{(n-1)})$
   (c) Compute $EMD_k \leftarrow$ EMD$(P_k^{(n)}, Q_0, d_k)$
   (d) Compute particle weight according to (17)
3. Find new target state $X_{Target}^{(n)}$ according to (18)
4. Build target signature $P_{Target}^{(n)}$ and ground distance $d_{Target}$
5. Compute EMD flow $f_{i,j} \leftarrow$ EMD$(P_{Target}^{(n)}, Q_0, d_{Target})$
6. Update parameters $\Theta^{(n)}$ according to (19).
7. Create new particle set $\{X_k^{(n+1)}\}_{k=1}^N$ using the Condensation algorithm  Isard and Blake (1998).
```

We use a Bayesian tracking formulation where the goal is to find the most probable state at frame $n$ denoted $X_{Target}^{(n)}$. This state in our case is a rectangle defined by $(x, y, w, h)$. Given some particle state at frame $n$, $X_k^{(n)}$ and the observations up to frame $n$ , $Z_k^{(1:n)}$, which are the signatures associated with that state. We assume that target dynamics form a temporal Markov chain so that $Pr(X_k^{(n)}|X_k^{(n-1)}, \ldots, X_k^{(1)}) = Pr(X_k^{(n)}|X_k^{(n-1)})$. We would like to estimate the posteriori probability $Pr(X_k^{(n)}|Z_k^{(1:n)})$. Using Bayesian formulation we have:

$$Pr(X_k^{(n)}|Z_k^{(1:n)}) = c_k^{(n)} \cdot Pr(Z_k^{(n)}|X_k^{(n)}) Pr(X_k^{(n)}|Z_k^{(1:n-1)}) \tag{15}$$

where

$$Pr(X_k^{(n)}|Z_k^{(1:n-1)}) = \int Pr(X_k^{(n)}|X_k^{(n-1)}) Pr(X_k^{(n-1)}|Z_k^{(1:n-1)}) dX_k^{(n-1)} \tag{16}$$

and $c_k^{(n)}$ is a normalization constant that does not depend on $X_k^{(n)}$. Computing equation (15) requires multiplying the observation density $Pr(Z_k^{(n)}|X_k^{(n)})$ by the effective prior $Pr(X_k^{(n)}|Z_k^{(1:n-1)})$. This effective prior term is generated based on the process dynamics which determine $Pr(X_k^{(n)}|X_k^{(n-1)})$ and using the posterior from the previous time step $Pr(X_k^{(n-1)}|Z_k^{(1:n-1)})$ which will be discussed later. The observation density in (15) is in fact the probability $Pr(P_k^{(n)}|Q_0, \Theta)$ inferred using LOM, where $P_k^{(n)}$ is the corresponding patch representation for the $k$'th particle in the $n$'th frame and $Q_0$ is our target patch representation. To define this conditional probability between patches we only have to define the probabilistic noise model for single pixels $Pr(p|q, \Theta)$. Then the ground distance for the EMD is defined as $d(p, q) = -log(p|q, \Theta)$ and $Pr(P_k^{(n)}|Q_0, \Theta)$ is obtained by solving the EMD problem.

Since solving an EMD problem can be a computationally challenging task, instead of using raw pixel values we work with superpixels. Specifically, we use  Levinshtein et al. (2009)

TurboPixels clustering algorithm to produce over segmentation in a region-of-intrest (ROI) which supports all the particle related patches. Candidate patches are then represented by signatures which are generated from this superpixel image. A signature consists of $M$ clusters that reside in the signature support, i.e. a rectangle. Each cluster is represented by its location , i.e. geometric center of mass (in patch canonical coordinates ranging $[0, 1]$), and average appearance (e.g. average $HSV$ values). We note that each signature is normalized to have unit weight. In this setup running LOM maps all the weight of the target patch to the candidate patch. This does not result in a 1:1 mapping of superpixles nor pixels, since target superpixels can be split into several candidate superpixels, also each patch might be comprised of a different number of pixels. This however does not pose a theoretical problem (see proposition 2) nor a practical one since solving the EMD optimization problem is still feasible and the flow obtained can still be used for parameter estimation.

The full Particle Filtering scheme we use follows closely the Condensation algorithm of Isard and Blake (1998). For each new frame that comes in we do the following: Build a signature for each of the $N$ particles according to their state i.e. each particle represents a rectangular image patch. Then calculate the EMD between each of these candidate signatures $\{P_k^{(n)}\}_{k=1}^N$ and the target signature $Q_0$ with ground distances as explained above (calculated using the noise model parameters $\varTheta$). The EMD scores $\{EMD_k^{(n)}\}_{k=1}^N$ are then used to set particle weights according to:

$$\pi_k^{(n)} = \frac{e^{-\beta \cdot EMD_k^{(n)}}}{\sum_{k=1}^N e^{-\beta \cdot EMD_k^{(n)}}} \tag{17}$$

We now have a set of weighted particle states $\{X_k^{(n)}, \pi_k^{(n)}\}_{k=1}^N$ with which we do two things. The first is compute the final state $X_{Target}^{(n)}$ according to:

$$X_{Target}^{(n)} = E[X^{(n)}] = \sum_{k=1}^{N} X_k^{(n)} \cdot \pi_k^{(n)} \tag{18}$$

The second thing we do with this weighted particle set is create a new particle set for the next frame that comes in. This is done as described in Isard and Blake (1998) by sampling particles from the current weighted set (with probability proportional to their weight) then subjecting them to a prediction phase according to the process dynamics which also include some random noise process. This factored sampling process uses $Pr(X_k^{(n-1)}|Z_k^{(1:n-1)})$ in order to produce a particle set approximating the required $Pr(X_k^{(n)}|Z_k^{(1:n-1)})$.

Finally the last step of each iteration is noise model parameter update based on the new target state found. This stage is carried out as follows. We begin by building $P_{Target}^{(n)}$ the signature for our new target state $X_{Target}^{(n)}$. Then we compute the EMD flow between this signature and the target signature $Q_0$ providing the most probable mapping between source and target signatures. Using this flow we estimate the noise distribution parameters $\varTheta_{ML}^{(n)}$ according to what is described in section 4. These estimated parameters are then regulated using a prior $\varTheta_{Prior}$ and a moving average (MA) process before producing the final parameters $\varTheta^{(n)}$:

$$\begin{aligned} \varTheta_{MAP}^{(n)} &= \tfrac{\varTheta_{ML}^{(n)} + \varTheta_{Prior} \cdot w_{Prior}}{1 + w_{Prior}} \\ \varTheta^{(n)} &= (1 - \alpha_{MA}) \cdot \varTheta^{(n-1)} + \alpha_{MA} \cdot \varTheta_{MAP}^{(n)} \end{aligned} \tag{19}$$

We emphasize that noise model parameters reflect the degree of rigidity in the target, thus on-line noise model parameter update enables LOT to adapt to different types of targets.

Although we do not update the target template explicitly, noise parameter update can be viewed as a limited form of template update. This is because setting noise model parameters directly affects the EMD solutions space thus effectively changing the space of possible matches.

We note that currently our algorithm does not handle occlusions explicitly, however partial occlusions are handled quite well due to two mechanisms: one is the use of particle filtering which allows locking back to a target after it was lost due to an occlusion. Additionally since in many cases the visible part of an occluded target shares its color statistics with the entire target, LOT can reduce to tracking only the visible part of the target, allowing it to overcome partial occlusions.

## 6 Experiments

This section presents experimental results. We begin with a synthetic experiment demonstrating noise parameter estimation in a simplified scenario. Then we present the experimental setup, parameter configurations, data-sets and performance evaluation method that will be used throughout the rest of the experimental section. We present the on-line parameter estimation of LOT on both a toy-example and real data, followed by an experiment illustrating the significance of updating the noise parameters on-line using benchmark sequences. We then consider two noise models and compare their performance after which we present quantitative results comparing LOT with 6 state-of-the-art tracking algorithms.

### 6.1 Synthetic Noise Model Parameter Estimation Experiment

The following synthetic experiment demonstrates parameter estimation for the Gaussian noise model in a simplified scheme. We estimate the noise in location $\sigma_L$ and the noise in appearance $\sigma_A$. This estimation is done in a Maximization-Maximization (MM) scheme where we solve the EMD then estimate the noise model parameter using an ML estimator, we then use these parameters to solve the EMD again and so on until both EMD and parameters converge. We note that convergence to a local maxima is guaranteed as both MM steps (i.e. EMD and ML) increase the likelihood.

Figure 1 shows the result of two such experiments. We started with an image over-segmented into super-pixels and coarsened it appropriately. This coarsened image was contaminated by two types of noise. In the first case additive-white-Gaussian-noise (AWGN), $\sigma_A = 0.4$, was added to the cluster appearances. $\sigma_A$ was also empirically calculated from the known cluster correspondence ($\sigma_A^{Empiric} = 0.417$). We performed parameter estimation using our MM scheme allowing both $\sigma_L$ and $\sigma_A$ to vary (starting from $\sigma_L = 0.05, \sigma_A = 0.05$) and found that $\sigma_A$ converged to $0.396$ . For better visualization of the results we colored each cluster of the noisy image according to the coarsened image cluster which contributed the maximal amount of weight to it based on the EMD flow. For comparison, we repeated the experiment with randomly chosen parameter values $\sigma_L = 0.35, \sigma_A = 0.11$ (drawn uniformly from the interval $[0, 0.5]$), and found that using the estimated parameters produces better visual results.

In the second case cluster appearances were locally permutated (i.e. cluster colors were swapped) to produce localization noise. This time $\sigma_L$ was empirically calculated using

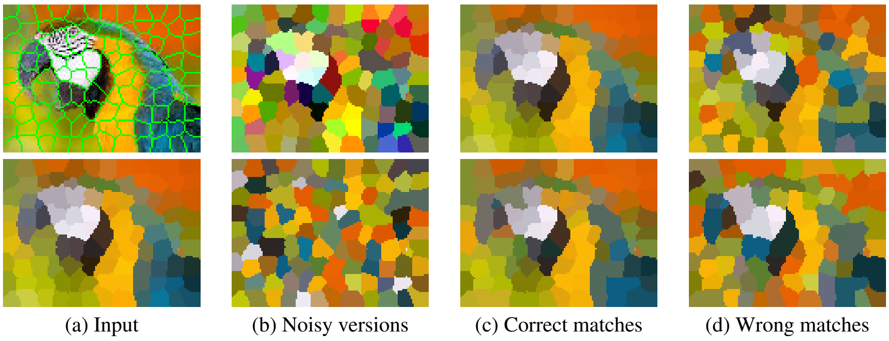


**Fig. 1** Synthetic Experiment : (a) Original image with super-pixel boundaries (top) and image coarsened into super pixels (bottom) (b) Coarsened image with additive-white-Gaussian-noise added to the color of the super pixels (top) and super-pixel appearance permutation (bottom). (c) Matching found of images in (b) to coarsened image using estimated $\sigma_A, \sigma_L$ (top/bottom accordingly). Colors projected from coarsened image. (d) Matching found of images in (b) to coarsened image using random $\sigma_A, \sigma_L$ (top/bottom accordingly). Colors projected from coarsened image.

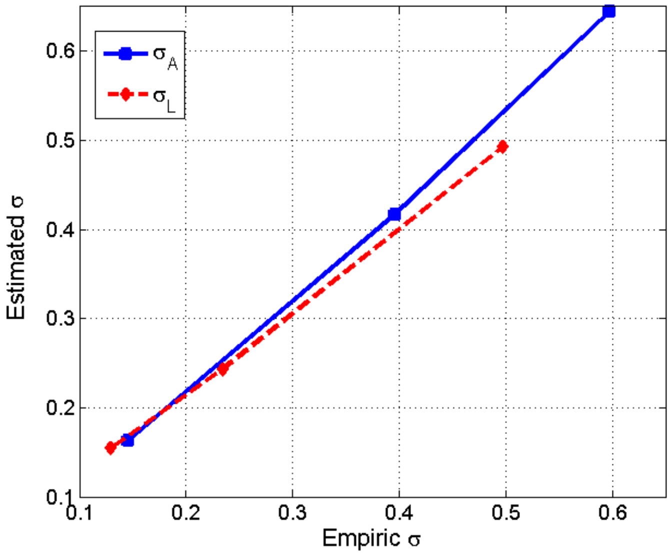


**Fig. 2** In blue: Values of $\sigma_A$ estimated using Maximization-Maximization (MM) vs. empirical values calculated based on known noise values. Contaminating noise was additive-white-Gaussian-noise (AWGN), with different variance values, added to cluster appearance. In red: $\sigma_L$ values estimated using MM vs. empirical values calculated based on known noise (permutation). Localization noise was modeled as local cluster appearance permutation i.e. swapping cluster colors in different neighborhood sizes.

the known correspondence ($\sigma_L^{Empiric} = 0.497$) and compared to our estimated parameters (same initialization) calculated without knowing the permutation. Again, our estimations converged correctly to the value $\sigma_L = 0.492$, taking $\sigma_A \to 0$ . Colors were projected as explained before to demonstrate the advantage over using random parameter values ($\sigma_L = 0.28, \sigma_A = 0.32$).

This experiment was repeated for 2 additional $\sigma_A$ and $\sigma_L$ values. Figure 2 presents the values of $\sigma_A, \sigma_L$ found using our MM estimation versus empirical values measured from the actual noises. The graphs show we can consistently estimate the parameters correctly for a wide range of noises.

We conclude that parameter estimation based on the EMD solution can produce results highly correlated with the contaminating noises at least in the case of the Gaussian noise model.

### 6.2 Experimental Setup

In the experiments reported here $HSV$ color space is used for appearance description. Both appearance and location spaces are used in a canonical form, i.e. normalized to the range $[0, 1]$. Cluster weights are determined according to the fraction of pixels associated with them in the signature (thus ensuring a total signature weight of 1). The state vector includes position and scale, i.e. $X_i = (x_i, y_i, w_i, h_i)$, and a zero-order motion model is assumed thus the process model for the Condensation algorithm includes only effects of noise. For $x$ and $y$ the process noise is additive-white-Gaussian-noise (AWGN) with $\sigma_{xy} = 7$. For scale parameters $w$ and $h$ we use multiplicative-Gaussian-noise with mean 1 and $\sigma_{wh} = 0.07$ (i.e. STD reflecting $7\%$ scale change). The noise is added to each of the state variables independently. We use $N = 250$ particles with particle weighing parameter $\beta$ set to 10 in order to better differentiate between particle scores.

Superpixels are built in a ROI that supports all the particles. The desired number of superpixels (a parameter of the Turbopixel algorithm) is set in the range $300 - 1000$ where the actual value is determined such that the target region would consist of roughly 20 superpixels. All the parameters are kept fixed for all experiments.

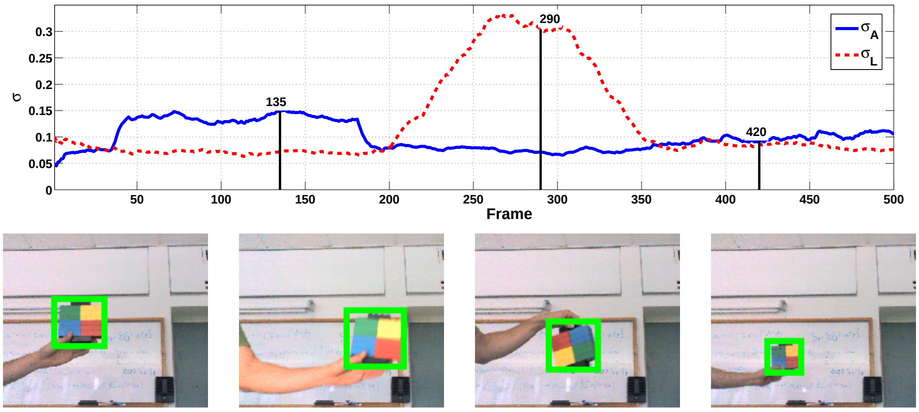
Using a standard PC equipped with an Intel Core i7 processor our Matlab-Mex implementation runs at $\sim 1$ frame per second for a target window size of $\sim 50 \times 50$ pixels. The run time is divided almost equally between two major time consuming operations which are the superpixel clustering and the EMD calculation for all the particles.

Our evaluation dataset is comprised of two subsets. The first includes 9 commonly used sequences (5 color and 4 grayscale) appearing in recent related publications (Ross et al. (2004, 2007); Babenko et al. (2009); Mei et al. (2011); Kwon and Lee (2010); Wang et al. (2011); Santner et al. (2010)). The second subset presents 6 challenging new sequences including gray-scale and color examples with both static and moving cameras. The targets in these sequences are subject to many appearance changes due to different types of deformations, pose changes, out-of-plane-rotations, massive scale changes, motion blur and illumination changes.

We adopt the widely used PASCAL VOC (Everingham et al. (2010)) criterion which quantifies both the centering accuracy as well as the scale accuracy. The criterion is $a_0 = \frac{area(B_p \cap B_{gt})}{area(B_p \cup B_{gt})}$ where $B_p$ and $B_{gt}$ denotes the predicted and ground truth bounding boxes accordingly. Successful tracking is considered as $a_0 > 0.5$ $(50\%)$. We note that some of the sequences were re-annotated in order to provide ground truth for each frame that also accounts for scale changes disregarded in some of the original annotations.

## 6.3 On-line Parameter Update

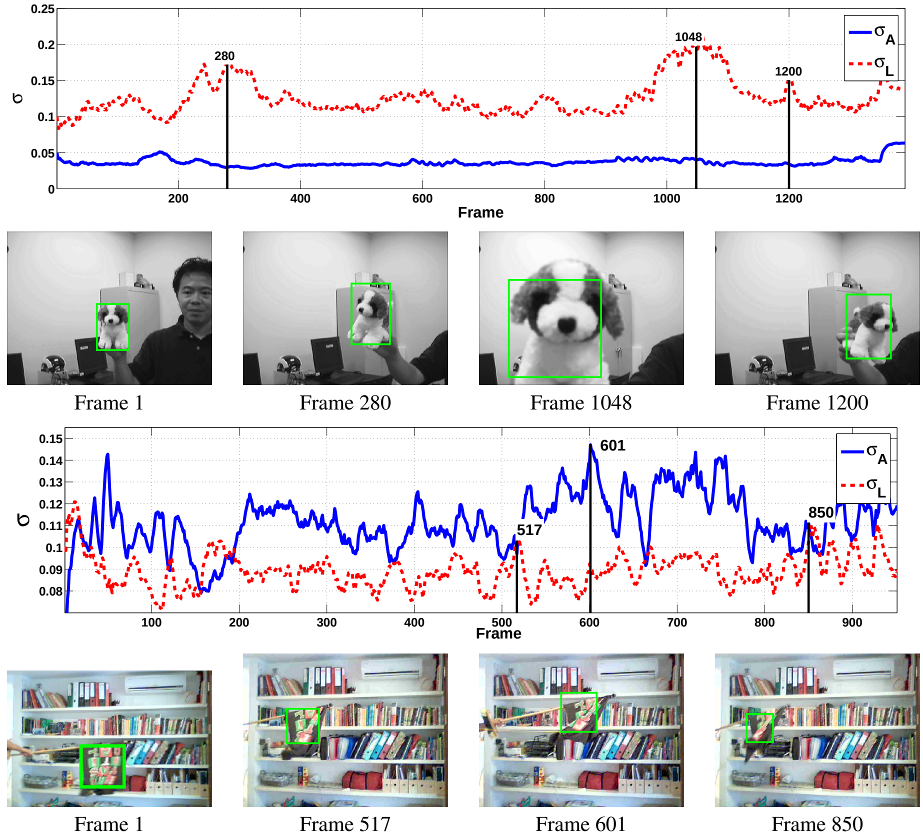
Before we start the performance evaluation on the main dataset we would like to first demonstrate the on-line noise parameter update capabilities of LOT. We begin with a $500$ frame toy-example of a LEGO target subject to both appearance and localization noises. We use the Gaussian noise model which means Gaussian noise for both appearance and localization (as presented in section 4.1). This Gaussian model has two parameters i.e. $\Theta = \{\sigma_A, \sigma_L\}$ which are updated according to (19). Parameters are initialized according to $\sigma_{A^{prior}} = 0.05, \sigma_{L^{prior}} = 0.1$, with prior weights set to $w_{\sigma_A^{prior}} = w_{\sigma_L^{prior}} = 0.25$. The ML estimators are calculated according to equation (14) and the MA parameter is fixed to $\alpha_{MA} = 0.3$. Figure 3 presents the behavior of the noise parameters $\sigma_L$ and $\sigma_A$ and sample frames for the LEGO sequence. The target is first subject to an illumination change. LOT detects the appearance change and increases the appearance noise parameter $\sigma_A$ while maintaining perfect tracking. As the illumination returns to normal the value of $\sigma_A$ decreases. Next the target is rotated about its origin. The tracker state space does not include rotation therefore this rotation is effectively localization noise. As before the target is tracked perfectly while LOT estimates and adapts the value of $\sigma_L$ on-line, increasing $\sigma_L$ as the rotation angle increases and then decreasing $\sigma_L$ as the target is rotated back. Towards the end of the sequence the algorithm correctly tracks target scale changes without altering the noise parameters which is a desired behavior.



Frame 1 Frame 135 Frame 290 Frame 420

**Fig. 3** Parameter estimation for the LEGO sequence: (Top) Noise parameter values, $\sigma_L$ (Dashed-Red) and $\sigma_A$ (Solid-Blue) per frame showing their on-line update.(Bottom) Four sample frames. First the target is illuminated with a strong light causing an appearance change handled by increasing $\sigma_A$. Next the target is rotated and since we only model $2D$ translation (w/o rotation) this creates localization noise which is handled by a large $\sigma_L$ variation. Finally the target moves away from the camera causing a scale change which is correctly tracked without significant noise parameter changes.

We proceed to demonstrate the on-line parameter update using two sequences from our dataset. Figure 4 presents the noise parameter values as well as sample frames for the Dog and Shirt sequences. For the Dog sequence (Top) we observe that as there are no illumination or other appearance changes the value of $\sigma_A$ remains low and almost constant throughout the entire sequence. The value of $\sigma_L$ on the other hand is updated due to localization noise



Frame 1 Frame 280 Frame 1048 Frame 1200

Frame 1 Frame 517 Frame 601 Frame 850

**Fig. 4** Parameter estimation for the Dog (upper half) and Shirt (bottom half) sequences demonstrating on-line noise parameter update. For each sequence, on top is Noise parameter values, $\sigma_L$ (Dashed-Red) and $\sigma_A$ (Solid-Blue) per frame showing their on-line update. On the bottom four sample frames. For the Dog sequence there are no appearance changes and mainly out-of-plane rotations which cause localization noise affecting only $\sigma_L$. For the Shirt we observe appearance noise dominates over localization noise explaining motion blur and local self occlusions.

in the form of target out-of-plane rotations and scale changes that push part of the target out of the frame support.

For the Shirt sequence (bottom part of figure 4) we notice that both noise model parameters are actively updated although $\sigma_A$ dominates over $\sigma_L$. The main reasons for this behavior are: 1) the rapid movement of the shirt creates motion blur mixing the different colors creating appearance changes. 2) The waving of the bottom part of the shirt cause local deformations and self occlusions that most of the time do not affect the top-left part of the target (which does not suffer localization changes). Due to these reasons LOM tends to associate matching errors with appearance changes rather than localization noise.

## 6.4 Significance of On-line Parameter Update

In order to point out the importance and significance of on-line parameter estimation we compare the performance of LOT with and without on-line parameter update. To do so we use the Gaussian noise model considering two fixed parameter configurations. In the first configuration we fix $\sigma_A = 0.05, \sigma_L = 0.2$ making the cost of changing cluster appearances more expensive then the cost of changing a clusters locations. In the second configuration we fix $\sigma_A = 0.2, \sigma_L = 0.05$ making appearance changes less costly compared with location changes. We evaluate the tracking performance as explained above using our full dataset (15 sequences).

**Table 1** Comparison of tracking performance with fixed noise parameters vs. on-line updating parameters. Table entries are the percent of frames for which the PASCAL criterion was $a_0 > 0.5$. Best result are in **bold** preface. As can be seen on-line parameter estimation improves tracking performance in 12 out of 15 sequences.

| Sequence | Dataset | $\sigma_A = 0.05, \sigma_L = 0.2$ | $\sigma_A = 0.2, \sigma_L = 0.05$ | On-line |
|---|---|---|---|---|
| Shop | Common | 33.6 | 33.6 | **34.6** |
| Girl | Common | 32.3 | 49.1 | **67.6** |
| Human | Common | 94.2 | 7.3 | **97.6** |
| Skating | Common | 13.4 | 15.6 | **29.4** |
| Lemming | Common | 67.4 | 56.4 | **73.8** |
| Dog | Common | 91.3 | 51.4 | **97.4** |
| David | Common | 8 | 3.5 | **10** |
| Sylv | Common | **48.7** | 2.6 | 46 |
| Face | Common | 42.5 | 21.7 | **44.4** |
| DH | New | 33.9 | 16.8 | **92.3** |
| Shirt | New | 75 | 60.9 | **88.1** |
| Train | New | 53.2 | **82** | 69.6 |
| UCSDPeds | New | 6.5 | 1.5 | **73.9** |
| Boxing | New | **72.8** | 55.7 | 70.1 |
| Towel | New | 97.3 | 93 | **99.7** |

Results presented in Table 1 demonstrate the significance of on-line parameter update. When using on-line update the algorithm outperforms the fixed parameter configuration in 12 out of 15 cases. For some of these sequences performance with on-line update is close to the best results produced by fixed parameters (e.g. Human and Face) while for other sequences on-line update allows better performance relative to both fixed parameter configuration (e.g DH, Skating and Girl) demonstrating the advantage of the on-line parameter updating scheme. Closely examining the remaining 3 examples where fixed parameters do better (Sylv,Train,Boxing), we can see that for Sylv and Boxing fixed parameters, with $\sigma_A = 0.05$ and $\sigma_L = 0.2$, only give a marginal absolute improvement of less than $3\%$. Thus only in one case (Train) fixed parameters, with $\sigma_A = 0.2$ and $\sigma_L = 0.05$, are able to produce substantially better results than on-line updating parameters ($82\%$ vs. $69.6\%$). In this specific case the performance gain is due to better behavior under several partial occlutions allowing the fixed parameters to retain large overlap while the updating parameters cause the target to shrink only to the visible part of the target. Overall, although on occasion fixed parameter configurations can lead to better performance, in general, fixed parameters lack the much needed flexibility to cope and explain different object types and different levels of target rigidity. On-line parameter update grants us this adaptation flexibility which leads to better tracking performance in most cases, many times exceeding what a single fixed parameter configuration can achieve.

### 6.5 Noise models

LOM derivation is general in the sense that any noise model can be plugged into this framework. The noise model controls the ground distance in the EMD optimization and therefore its choice can have a substantial affect on tracking performance especially if the noise used does not model the true nature of the noise domain.

We experiment with two noise models based on our derivations from section 4 and appendix A. The first is the Gaussian model, already presented, where both appearance and localization noises are Gaussian. The initialization configuration for this model is as explained in section 6.3. The second noise model we test is a Gaussian-Uniform model where the appearance noise is Gaussian while the localization noise is a mixture-of-uniforms (as discussed in appendix A.2). This model has 3 parameters $\Theta = \{\sigma_A, r, \alpha\}$. Appearance noise variance $\sigma_A$ was initialized as in the Gaussian case. The mixture-of-uniforms parameters $r, \alpha$ were set according to the following priors $r_{prior} = 0.2$ and $\alpha_{prior} = 0.9$ with all prior weights set to $0.25$ as before.

**Table 2** Tracking using different noise models. Comparing performance with two different noise models the Gaussian-Uniform and Gaussian. Table entries are the percent of frames for which the PASCAL criterion was $a_0 > 0.5$. Best result are in **bold** preface. Results suggest that the Gaussian noise model is better suited for modeling the noises at hand and their parameter domain.

| Sequence | Dataset | Gaussian-Uniform | Gaussian-Gaussian |
|---|---|---|---|
| Shop | Common | 33.8 | **34.6** |
| Girl | Common | 16.6 | **67.6** |
| Human | Common | 94.2 | **97.6** |
| Skating | Common | 11.2 | **29.4** |
| Lemming | Common | **76.3** | 73.8 |
| Dog | Common | 41.1 | **97.4** |
| David | Common | 4.5 | **10** |
| Sylv | Common | 45.7 | **46** |
| Face | Common | 36.3 | **44.4** |
| DH | New | 54.9 | **92.3** |
| Shirt | New | 76.8 | **88.1** |
| Train | New | 4.12 | **69.6** |
| UCSDPeds | New | 71.3 | **73.9** |
| Boxing | New | **71.5** | 70.1 |
| Towel | New | 98.1 | **99.7** |

Tracking performance using both noise models is presented in Table 2. We see that the Gaussian model outperforms the Gaussian-Uniform model in almost every case. Even when the Gaussian-Uniform model provides better tracking performance (i.e. Lemming and Boxing) the improvement is only marginal. These results demonstrate the importance of noise model selection and its effect on tracking performance. Evidently, choosing an adequate noise model and parametrization can enhance tracking performance.

In light of these results the rest of our experiments are conducted using the Gaussian model.

## 6.6 Comparison with other tracking algorithms

We compare LOT's performance with 7 state-of-the-art tracking algorithms with publicly available implementations: Incremental Visual Tracking (IVT) Ross et al. (2007), Online AdaBoost (OAB) Grabner et al. (2006), Multiple Instance Learning (MIL) Babenko et al. (2009), Visual Tracking Decomposition (VTD) Kwon and Lee (2010), Tracking-Learning-Detection (TLD) Kalal et al. (2010) , Robust Object Tracking via Sparsity-based Collaborative Model (SCM) Zhong et al. (2012) and Visual Tracking via Adaptive Structural Local Sparse Appearance Model (ASLA) Jia et al. (2012).

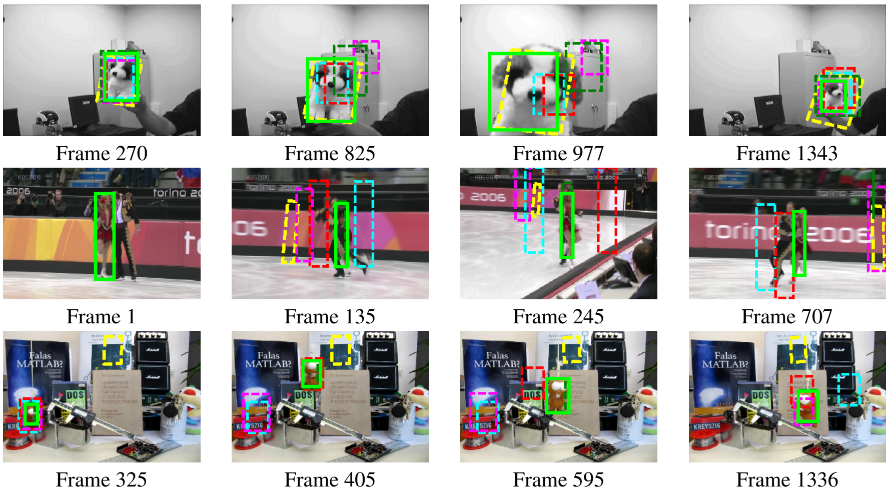
### *6.6.1 Commonly Used Sequences*

Quantitative results for the commonly used sequences data-set are presented in Table 3, where it can be seen that LOT exhibits performance comparable to the state-of-the-art algorithms producing best performance in 3 out of 9 sequences (Human, Skating, Lemming) and ranks second in 2 additional sequences (Dog and Girl) .

**Table 3** Quantitative comparison, for 9 commonly used sequences, showing the percent of frames for which the PASCAL criterion was $a_0 > 0.5$. Best result are in **bold** preface.It can be seen that LOTs performance is comparable to the state-of-art algorithms.

| Sequence | RGB / GL | IVT | OAB | MIL | VTD | TLD | SCM | ASLA | LOT |
|---|---|---|---|---|---|---|---|---|---|
| Shop | RGB | 36.4 | 20.9 | 20.9 | 35 | 23.9 | **100** | 36.4 | 34.6 |
| Girl | RGB | 15.4 | 26.2 | 25 | **93.4** | 54.5 | 34.5 | 16.6 | 67.6 |
| Human | RGB | 88.8 | 26.2 | 25 | 64.6 | 75 | 92.5 | **97.1** | **97.6** |
| Skating | RGB | 3.8 | 8.8 | 9.8 | 11.5 | 4.1 | 11.9 | 5.1 | **29.4** |
| Lemming | RGB | 16.2 | 37.1 | 37.6 | 54.3 | 25.2 | 16.6 | 16.8 | **73.8** |
| Dog | GL | 87 | 57 | 45.5 | 70 | 30.4 | 82.3 | **100** | 97.4 |
| David | GL | 83.1 | 9.7 | 19.3 | 18.8 | 63.4 | **100** | 34.2 | 10 |
| Sylv | GL | 45.7 | 31 | 73.2 | **93.4** | **93.5** | 78.1 | 56.2 | 46 |
| Face | GL | **99** | 75 | 54.5 | 70.1 | **99.3** | 88.1 | 94.4 | 44.4 |

When we examine sequences where LOT experiences difficulties we find that: For the Shop sequence all methods, except SCM which has an occlusion handling scheme, lose track when the target is occluded at around frame 200. It can be seen that IVT, VTD and LOT (which do not handle occlusions explicitly) produce almost identical results in this case ($36.4\%, 35\%$, and $34.6\%$ accordingly), and ASLA which has an occlusion handling mechanism produces similar results as these methods. In the Girl sequence the target (a girls head) makes a 360-degree out-of-plane rotation. At the point where the girls face is completely occluded and only the hair is visible LOT looses track and drifts to a background object. In its current form LOT does not change the object template and although our noise model parameter update can be viewed as a form of constrained template updating it is still insufficient for handling long and full occlusion such as the one experienced in the Girl sequence. The remaining 3 sequences (David, Sylv and Face) are gray-scale sequences. LOT can run in both color and gray-scale (e.g. Dog), however using color appearance representation (i.e. $HSV$) makes gray-scale more challenging as it leaves the algorithm with only a single appearance channel. This makes coping with severe global and local illumination changes a difficult challenge and it is mainly for this reason that LOT's performance degrades on the last 3 sequences where it ranks last in David, fifth in Sylv and last in Face. Although some



Frame 270 Frame 825 Frame 977 Frame 1343

Frame 1 Frame 135 Frame 245 Frame 707

Frame 325 Frame 405 Frame 595 Frame 1336

**Fig. 5** Sample frames from three sequences: Dog, Skating and Lemming. The different algorithms are: IVT in Yellow, OAB in Cyan, MIL in Red, VTD in Magenta and LOT in light Green.

methods produce better results for some sequences, looking at the entire dataset it can be seen that the overall performance of LOT is comparable to the state-of-the-art methods. We believe that MIL and OAB have poorer performance mainly due to their lack of scale adaptability.

Figure 5 presents sample frames from 3 sequences (Dog, Skating and Lemming) qualitatively showing LOT's ability to cope with difficult appearance changes such as massive scale changes and out-of-plane-rotations.

### *6.6.2 New Sequences*

LOT was also compared to the state-of-the-art algorithms using our second data-set of 6 challenging new sequences. Sample frames from these sequences are presented in Figure 6.

The first, 481 frame long, sequence shows a Down-Hill (DH) bike ride. As the rider jumps and moves in and out of shade a lot of motion blur, deformations and illumination changes are created. IVT, TLD and SCM drift after the first jump at around frame 56, MIL and OAB keep tracking until around frame 400 where they also drift. Only VTD and LOT are able to track the rider until the end of the sequence.

The second, 951 frame long, sequence we captured is of a T-shirt undergoing severe non-rigid deformations and motion blur. LOT with its inherent ability to explain non-rigid deformations is able to track the shirt throughout the entire length of the sequence. LOT outperforms all other methods except ASLA which is able to produce more accurate results.

The third, 900 frame long, sequence was taken from the PETS-2006 dataset[1]. It shows a man walking around a busy train station making many pose changes and undergoing several occlusions. Although LOT does not have an explicit mechanism for handling occlusions it is able to handle partial occlusions by tracking the remaining visible part of the target which often captures the full target color statistics. In this sequence the first partial occlusion occurs around frame 35 causing IVT, OAB, TLD, ASLA and MIL to drift. A second occlusion at

[1] http://www.cvg.rdg.ac.uk/PETS2006/data.html

**Table 4** Quantitative comparison, for 6 new sequences, showing the percent of frames for which the PASCAL criterion was $a_0 > 0.5$. Best result are in **bold** preface. LOT has the best overall performance on this dataset.

| Sequence | RGB / GL | IVT | OAB | MIL | VTD | TLD | SCM | ASLA | LOT |
|---|---|---|---|---|---|---|---|---|---|
| DH | RGB | 8.9 | 47.8 | 45.5 | 69.4 | 10.8 | 8.3 | 8.7 | **92.3** |
| Shirt | RGB | 0.5 | 66.7 | 32.5 | 79 | 0.6 | 80.8 | **92.4** | 88.1 |
| Train | RBG | 2.7 | 3.4 | 2.3 | 2.9 | 2.8 | 10 | 4.6 | **69.6** |
| UCSDPeds | GL | 26.4 | 42.5 | 26.8 | 60.5 | 55.6 | **100** | **99.2** | 73.9 |
| Boxing | RGB | 7.3 | 18.7 | 18.4 | 21.2 | 28.8 | 34.8 | 55.2 | **70.1** |
| Towel | RGB | 8.8 | 5 | 46 | 34.5 | 94.7 | 90.1 | 44 | **99.7** |

around frame 60 throws VTD and SCM off track as well. LOT is able to overcome these 2 occlusion by shrinking and matching to the remaining visible part of the target. It continues tracking the man for the entire length of the sequence while overcoming pose changes and additional occlusions. We note that the final tracking score for this sequence is only $69.7\%$ since during the occlusions the predicted bounding box shrinks to the visible part of the target while the ground truth annotation continues marking the whole occluded target.

The forth, $261$ frame long, gray-scale, sequence taken from the UCSD crowd dataset[2] shows two people walking and fighting. We track both people as a single target. This crowd target undergoes non-rigid deformations as the people draw nearer and apart and as they fight with each other. All the methods are able to track the targets location throughout most of the sequence with only minor glitches, LOT ranks in 3rd place producing results better than most trackers and second only to SCM and ASLA.

The fifth, $352$ frame long, is a boxing sequence. At the beginning of this sequence only LOT is able to correctly track the boxer through the difficult pose changes. All methods drift between frame 200-225 due to a rapid movement followed by a full occlusion however LOT is able to lock back on at frame 261 and continue tracking the target until the end of the sequence.

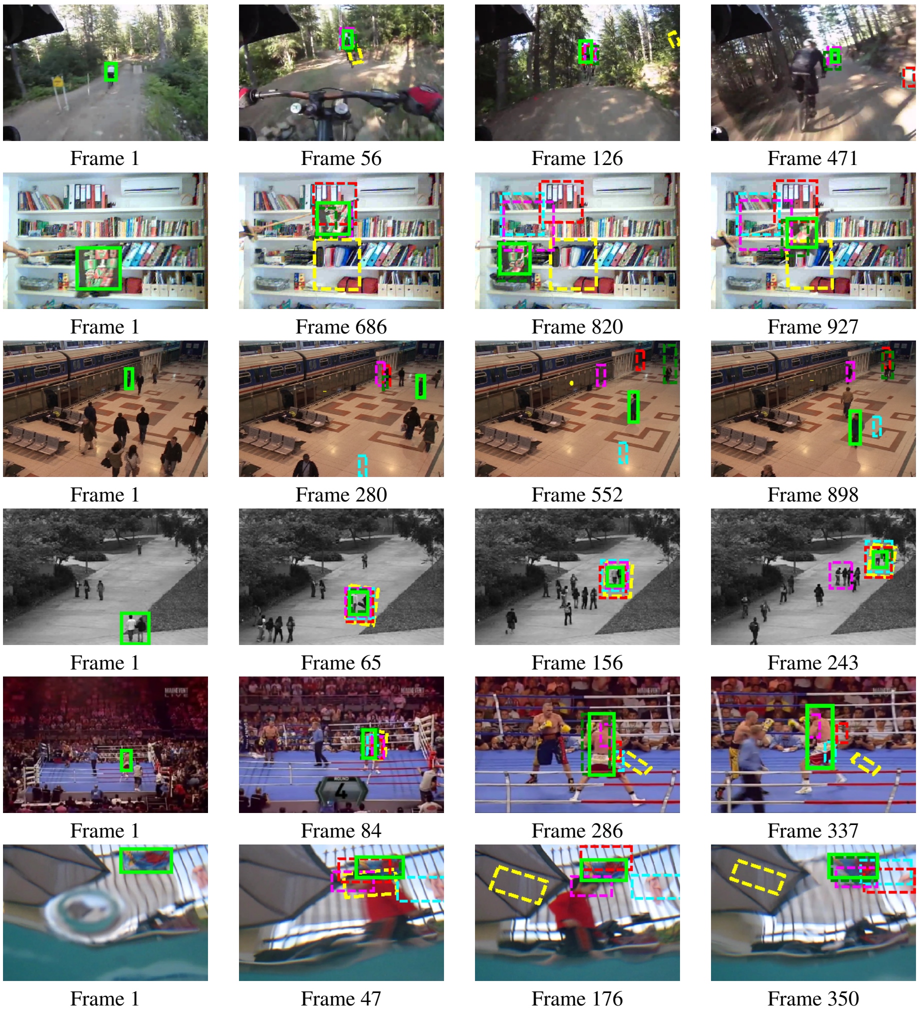
The sixth and last, $374$ frame long, sequence (taken from Alterman et al. (2012)) was shot from underwater into air. Our target is a towel hanging on a fence. This target is subject to a complex non-uniform deformation field caused by the waters movement. Due to the inherent properties of LOT it is able to handle these difficult deformations and produce nearly perfect tracking. TLD and SCM also produce good results for this sequence but not as good as LOT. Other algorithms drift.

Quantitative results are presented in Table 4. LOT has the best overall performance, it outperforms the other tracking methods producing better results for all but one sequence (where it comes second).

## 7 Conclusions

Locally Orderless Tracking is a new visual tracking algorithm that estimates and adapts, on-line, to the rigidity of the tracked object. The algorithm is governed by a small set of parameters $\Theta$ that are estimated on-line allowing it to go from rigid template matching on one end to histogram-like tracking on the other, or be anywhere in between. At the heart of this framework lies Locally Orderless Matching, a new probabilistic interpretation of EMD that rigorously shows how EMD can be used to infer the likelihood that patch $P$ is a noisy replica of patch $Q$ using some noise model with parameters $\Theta$. Since the framework

[2] http://www.svcl.ucsd.edu/projects/peoplecnt/index.htm


Frame 1 Frame 56 Frame 126 Frame 471

Frame 1 Frame 686 Frame 820 Frame 927

Frame 1 Frame 280 Frame 552 Frame 898

Frame 1 Frame 65 Frame 156 Frame 243

Frame 1 Frame 84 Frame 286 Frame 337

Frame 1 Frame 47 Frame 176 Frame 350

**Fig. 6** Sample frames from the new sequence set: DH, Shirt, Train, UCSDPeds, Boxing and Towel. The different algorithms are: IVT in Yellow, OAB in Cyan, MIL in Red, VTD in Magenta and LOT in light Green.

is generic any noise model can be plugged in and we have demonstrated the use of two such noise models. We have shown the significance and importance of estimating noise model parameters on-line and demonstrated how this parameter estimation and adaptation can be achieved using the data at hand both theoretically and empirically. Finally we have shown that LOT's performance is comparable to state-of-the-art methods on a wide range of commonly used and new videos presenting superior performance in many cases.

Future work is intended in 3 main aspects: The first is exploiting the flow produced during the EMD calculation not merely for parameter update but also for on-line template update and foreground-background target segmentation which can help in overcoming occlusions. The second aspect is speed where we believe that using different Superpixel schemes and maybe some EMD approximations can make this algorithm run at real-time. The third aspect is looking into different noise models and appearance representations that might be better suited for specific applications (such as tracking in gray-scale sequences).

## A Additional noise models

### A.1 Uniform Noise

A Uniform distribution with parameter $r$ can be used as location and/or appearance noise model again. Due to the independence assumed between appearance and location parameters $p, q, r, D$ will be used without the superscripts $A, L$.

$$Pr(p|q,r) = \begin{cases} \frac{1}{(2r)^D} & ||p-q||_\infty \leq r \\ 0 & \text{otherwise} \end{cases} \tag{20}$$

Where $D$ is the dimension of $p$ and $q$. The ground distance in this case is:

$$d(p,q) = \begin{cases} D \cdot log(2r) & ||p-q||_\infty \leq r \\ \infty & \text{otherwise} \end{cases} \tag{21}$$

This distance means the cost of changing the appearance and/or location of a pixel by less than a certain quant costs nothing (the same as not moving it at all), and changing it by more than that is not allowed.

This model may pose some problems as certain mismatches are not allowed at all and also since the signature EMD problem can become unfeasible in some cases i.e. giving $\infty$ distance. Therefore a mixture of two uniforms might be a better choice.

### A.2 Uniform-Mixture Noise

Using a mixture of two uniforms provides us with one low cost for small perturbations and a second high cost (but not $\infty$) for large ones.This means we allow any match but with high cost. The parameter for the second uniform should include the entire space. We formulate this model using a mixture variable $h \sim Bernoulli(\alpha)$ and marginalizing over it:

$$Pr(p|q,r,\alpha) = \alpha Pr(p|q,h=0) + (1-\alpha)Pr(p|q,h=1) \tag{22}$$

Where $P(p|q,h=\{0,1\})$ are both uniform distributions. The ground distance is given by:

$$d(p,q) = \begin{cases} -log(\frac{\alpha}{(2r)^D} + \frac{1-\alpha}{S}) & ||p-q||_\infty \leq r \\ -log(\frac{1-\alpha}{S}) & \text{otherwise} \end{cases} \tag{23}$$

Where $S$ is the hyper-volume of the entire space (e.g. for un-normalized RGB space $S = (2^8)^3$ which is the RGB cube volume).

#### *A.2.1 Uniform-Mixture Parameter Estimation*

This model has two parameters $\Theta = \{\alpha, r\}$. We use the EMD correspondence mapping $f_{ij}$ and the ground distance matrix $d_{ij} = d(p_i, q_j)$ from which we build a CDF of the transported distance. We denoted this CDF by $c(r) : [0,R] \rightarrow [0,1]$ where $R$ is the maximal distance a mass can move in our subspace i.e. $\forall r \quad c(r) = \frac{\sum\limits_{i,j:d_{ij} \leq r} f_{ij} \dot{d}_{ij}}{\sum\limits_{i,j} f_{ij} \dot{d}_{ij}}$. We can now estimate $\alpha$ and $r$ using an ML consideration:

$$\begin{aligned} &logPr(P|Q,r,\alpha) = \sum_i logPr(p_i|q_j) = \sum_{i \in D_1} log(\tfrac{\alpha}{(2r)^D} + \tfrac{1-\alpha}{S}) + \sum_{i \in D_2} log(\tfrac{1-\alpha}{S}) = \\ &N[c(r) \cdot log(\tfrac{\alpha}{(2r)^D} + \tfrac{1-\alpha}{S}) + (1-c(r)) \cdot log(\tfrac{1-\alpha}{S})] \end{aligned} \tag{24}$$

where $D_1 = \{i : ||p_i - q_j||_\infty \leq r\}, D_2 = \{i : ||p_i - q_j||_\infty > r\}$ and $N$ is the total mass. If we only want to estimate $r$ and leave $\alpha$ constant we can numerically find $r$ that maximizes (24). For estimating both $r$ and $\alpha$ we differentiate (24) with respect to $\alpha$ and compare to 0 which leads to:

$$\alpha = \frac{c(r)S - (2r)^D}{S - (2r)^D} \tag{25}$$

Plugging this result back to equation (24) we see that we need to find:

$$\underset{r}{argmax} \left( c(r) \cdot log(\frac{c(r)}{(2r)^D}) + (1 - c(r)) \cdot log(\frac{1 - c(r)}{S - (2r)^D}) \right) \tag{26}$$

Equation (26) can be solved numerically given $c(r)$ built using the EMD result and then $\alpha$ is calculated based on equation (25).

## B Proof of Proposition 2

*Proof* For all $i, j$ in (7), we take all the variables $\{f_{k_1 j}, \ldots, f_{k_{w_i^p} j}\}$ that correspond to $w_i^p$ similar pixels (with singleton weights). We then collapse each set into a single variable representing their sum $g_{ij} = \sum_{l=1}^{w_i^p} f_{k_l j}$. This can be done as their coefficients $(d_{k_l j})$ in the optimization argument $\sum\limits_{ij} f_{ij} d_{ij}$ are the same. Thus the $w_i^p$ constraints of the form $\sum_j f_{k_l j} = 1$ can be replaced with a single constraint demanding $\sum_j g_{ij} = w_i^p$ and the $w_j^q$ constraints of the form $\sum_i f_{i k_l} = 1$ can be replaced with a single constraint demanding $\sum_i g_{ij} = w_j^q$. We then obtain the following integer linear program (ILP):

$$\begin{aligned} & \min \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} g_{ij} d_{ij} \\ \text{such that} & \\ & \sum_{i=1}^{n_1} g_{ij} = w_i^p, \sum_{j=1}^{n_2} g_{ij} = w_j^q, g_{ij} \in \{0, 1, \ldots, \min(w_i^p, w_j^q)\} \end{aligned} \tag{27}$$

By construction we have that the space of feasible solutions w.r.t to optimization problem (7) did not change i.e. $\min \sum\limits_{i=1}^{m} \sum\limits_{j=1}^{m} f_{ij} d_{ij} = \min \sum\limits_{i=1}^{n_1} \sum\limits_{j=1}^{n_2} g_{ij} d_{ij}$ where the $d_{ij}$ on the left and right side of the equation are set according to the appropriate source and sink nodes. Again this is true since every $g_{ij}$ is simply a sum of $f_{ij}$ having the same ground distance $d_{ij}$. If we now write (27) in the canonical form (as we did in proposition 1) we see that the matrix $A$ is again totally unimodular which means that the relaxed linear programming (LP) problem has an integral solution. This relaxed LP is exactly optimization problem (8) and given a solution (i.e. the $g_{ij}$) to this problem we can always find an assignment to the $f_{ij}$ such that would satisfy (7). This is true since we can always break down the compact signatures back into the pixel-wise problem with singleton bins which as we have shown would have the same minima. □

## C Proof of Proposition 3

*Proof* It is enough to look at a single step of uniting two clusters. Assume we unite $p_{n_1}, p_{n_1 - 1}$ into a single cluster $\hat{p}_{n-1}$. For weight/flow assignment $f_{ij}$ we have:

$$\sum_{i=1}^{n_1} \sum_{j=1}^{n_2} f_{ij} d_{ij} = \sum_{i=1}^{n_1 - 2} \sum_{j=1}^{n_2} f_{ij} d_{ij} + \sum_{j=1}^{n_2} f_{n-1,j} d(p_{n_1 - 1}, q_j) + f_{n_1, j} d(p_{n_1}, q_j) \tag{28}$$

Denoting $C = \sum_{i=1}^{n_1-2} \sum_{j=1}^{n_2} f_{ij} d_{ij}$ and using the triangle inequality we have:

$$\begin{aligned} & C + \sum_{j=1}^{n_2} f_{n-1,j} d(p_{n_1-1}, q_j) + f_{n_1,j} d(p_{n_1}, q_j) \\ & \leq C + \sum_{j=1}^{n_2} f_{n_1-1,j} [d(p_{n_1-1}, \hat{p}_{n_1-1}) + d(\hat{p}_{n_1-1}, q_j)] + f_{n,j} [d(p_{n_1}, \hat{p}_{n_1-1}) + d(\hat{p}_{n_1-1}, q_j)] \end{aligned} \tag{29}$$

Reorganizing the last expression by collecting elements related to the distance between the original clusters and their crude version and elements related to the distance between the crude cluster and its assignment leads to,

$$\begin{aligned} & C + \sum_{j=1}^{n_2} (f_{n_1-1,j} + f_{nj}) d(\hat{p}_{n_1-1}, q_j) + w_{n_1-1} d(p_{n_1-1}, \hat{p}_{n_1-1}) + w_{n_1} d(p_{n_1}, \hat{p}_{n_1-1}) \\ & = C + \sum_{j=1}^{n_2} \hat{f}_{n_1-1,j} d(\hat{p}_{n_1-1}, q_j) + w_{n_1-1} d(p_{n_1-1}, \hat{p}_{n_1-1}) + w_{n_1} d(p_{n_1}, \hat{p}_{n_1-1}) \end{aligned} \tag{30}$$

Where $\hat{f}_{n_1-1} = f_{n_1-1} + f_{n_1}$. The expression $\sum_{i=1}^{n_1-2} \sum_{j=1}^{n_2} f_{ij} d_{ij} + \sum_{j=1}^{n_2} \hat{f}_{n_1-1,j} d(\hat{p}_{n_1-1}, q_j)$ appearing in the last line is the optimization argument $EMD(\widehat{P}, Q, d)$. Lets fix now the variables $\{f_{ij}\}_{i=1}^{n-2}, \hat{f}_{n_1-1}$ to the argmin values of the problem (the values achieving the minimun for $EMD(\widehat{P}, Q, d)$. Now using the inequality in (30) we have

$$\begin{aligned} & EMD(\widehat{P}, Q, d) = \sum_{i=1}^{n_1-2} \sum_{j=1}^{n_2} f_{ij} d_{ij} + \sum_{j=1}^{n_2} \hat{f}_{n_1-1,j} d(\hat{p}_{n_1-1}, q_j) \\ & \geq \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} f_{ij} d_{ij} - w_{n_1-1} d(p_{n_1-1}, \hat{p}_{n_1-1}) - w_n d(p_{n_1}, \hat{p}_{n_1-1}) \\ & \geq \operatorname{argmin}_{f_{ij}} \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} f_{ij} d_{ij} - w_{n_1-1} d(p_{n_1-1}, \hat{p}_{n_1-1}) - w_n d(p_{n_1}, \hat{p}_{n_1-1}) \\ & = EMD(P, Q, d) - w_{n_1-1} d(p_{n_1-1}, \hat{p}_{n_1-1}) - w_n d(p_{n_1}, \hat{p}_{n_1-1}) \end{aligned} \tag{31}$$

Since $w_{n_1-1} d(p_{n_1-1}, \hat{p}_{n_1-1}) + w_n d(p_{n_1}, \hat{p}_{n_1-1} > 0$ it follows that,

$$|EMD(P, Q, d) - EMD(\widehat{P}, Q, d)| \geq w_{n_1-1} d(p_{n_1-1}, \hat{p}_{n_1-1}) + w_{n_1} d(p_{n_1}, \hat{p}_{n_1-1}) \tag{32}$$

In an analogous way it can be shown that,

$$|EMD(P, Q, d) - EMD(P, \widehat{Q}, d)| \geq w_{n_2-1} d(q_{n_2-1}, \hat{q}_{n_2-1}) + w_{n_2} d(q_{n_2}, \hat{q}_{n_2-1}) \tag{33}$$

The proposition follows by repeating this argument for all $\hat{p}_i, \hat{q}_j$

## References

M. Alterman, Y. Schechner, P. Perona, J. Shamir, Independent components in dynamic refraction. CCIT Report 805 (2012)

S. Avidan, Ensemble Tracking, in *CVPR*, 2005, pp. 494–501

B. Babenko, M.H. Yang, S. Belongie, Visual Tracking with Online Multiple Instance Learning, in *CVPR*, 2009

S. Boltz, F. Nielsen, S. Soatto, Earth Mover Distance on Superpixels, in *ICIP*, 2010

D. Comaniciu, Bayesian Kernel Tracking., in *DAGM-Symposium*, 2002, pp. 438–445

A. Doucet, N. de Freitas, N. Gordon, Sequential monte carlo methods in practice. Springer-Verlag (2001)

A. Elgammal, R. Duraiswami, L.S. Davis, Probabilistic Tracking in Joint Feature-spatial Spaces, in *CVPR*, 2003

M. Everingham, L.J. Van Gool, C.K.I. Williams, J.M. Winn, A. Zisserman, The pascal visual object classes (voc) challenge. IJCV (2010)

B.V. Ginneken, B.M.T. Haar Romeny, Applications of Locally Orderless Images, in *Scale-Space Theories in Computer Vision*, 1999

M. Godec, P.M. Roth, H. Bischof, Hough-based Tracking of Non-rigid Objects, in *ICCV*, 2011

H. Grabner, M. Grabner, H. Bischof, Real-time Tracking Via Online Boosting, in *BMVC*, 2006

G.D. Hager, P.N. Belhumeur, Efficient region tracking with parametric models of geometry and illumination. PAMI (1998)

X. He, R. Zemel, D. Ray, Learning and Incorporating Top-down Cues in Image Segmentation, in *ECCV*, 2006

I. Heller, C.B.G. Tompkins, An extension of a theorem of dantzig's. Linear Inequalities and Related Systems, Annals of Mathematics Studies, 38, Princeton (NJ), 247–254 (1956)

D. Hoiem, A. Efros, M. Hebert, Geometric Context from a Single Image, in *ICCV*, 2005

M. Isard, A. Blake, Condensation - conditional density propagation for visual tracking. IJCV (1998)

X. Jia, H. Lu, M.H. Yang, Visual Tracking Via Adaptive Structural Local Sparse Appearance Model, 2012

Z. Kalal, K. Mikolajczyk, J. Matas, Tracking-learning-detection. TPAMI (2010)

J.J. Koenderink, A.J. Van Doorn, The structure of locally orderless images. IJCV (1999)

J. Kwon, K.M. Lee, Visual Tracking Decomposition, in *CVPR*, 2010

E. Levina, P. Bickel, The Earth Mover's Distance Is the Mallows Distance: Some Insights from Statistics, in *ICCV*, 2001

A. Levinshtein, A. Stere, K.N. Kutulakos, D.J. Fleet, S.J. Dickinson, K. Siddiqi, Turbopixels: Fast superpixels using geometric flows. TPAMI (2009)

X. Mei, H. Ling, Y. Wu, E. Blasch, L. Bai, Minimum Error Bounded Efficient L1 Tracker with Occlusion Detection, in *ICCV*, 2011

S. Oron, A. Bar-Hillel, D. Levi, S. Avidan, Locally Orderless Tracking, in *CVPR*, 2012

S. Peleg, M. Werman, H. Rom, A unified approach to the change of resolution: Space and gray-level. TPAMI (1989)

X. Ren, J. Malik, Learning a Classification Model for Segmentation, in *ICCV*, 2003

D. Ross, J. Lim, M.H. Yang, Adaptive Probabilistic Visual Tracking with Incremental Subspace Update, in *ECCV*, 2004

D. Ross, J. Lim, R.S. Lin, M.H. Yang, Incremental learning for robust visual tracking. IJCV (2007)

Y. Rubner, C. Tomasi, L.J. Guibas, The earth mover's distance as a metric for image retrieval. IJCV (2000)

J. Santner, C. Leistner, A. Saffari, T. Pock, H. Bischof, Prost:parallel Robust Online Simple Tracking, in *CVPR*, 2010

S. Wang, H. Lu, F. Yang, M.H. Yang, Superpixel Tracking, in *ICCV*, 2011

A. Yilmaz, O. Javed, M. Shah, Object tracking: A survey. ACM Comput. Surv. (2006)

Q. Zhao, Z. Yang, H. Tao, Differential earth mover's distance with its applications to visual tracking. PAMI (2010)

W. Zhong, H. Lu, M.H. Yang, Robust Object Tracking Via Sparsity-based Collaborative Model, in *CVPR*, 2012