UNSUPERVISED SPATIOTEMPORAL VIDEO CLUSTERING
A VERSATILE MEAN-SHIFT FORMULATION ROBUST TO TOTAL OBJECT OCCLUSIONS

Simon Mure, Thomas Grenier, Hugues Benoit-Cattin

Université de Lyon, CREATIS
CNRS UMR5220 ; Inserm U1044
INSA-Lyon ; Université Lyon 1, France

ABSTRACT

In this paper, we propose a mean-shift formulation allowing spatiotemporal clustering of video streams, and possibly extensible to other multivariate evolving data. Our formulation enables causal or omniscient filtering of spatiotemporal data, which is robust to total object occlusions. It embeds a new clustering algorithm within the filtering procedure that will group samples and reduce their number over the iterations. Based on our formulation, we express similar approaches and assess their robustness on real video sequences.

Index Terms— Unsupervised spatiotemporal filtering, Mean-shift, Video clustering, Total object occlusion

1. INTRODUCTION

The mean-shift (M-S) technique, originally proposed by [1] is widely used in the context of image filtering, image segmentation [2] and tracking [3, 4]. When applied to video streams these methods are only able to process data frame by frame, which leads to a lack of temporal coherence between the filtered results.

As far as we know [5] were first to briefly describe how to extend the mean-shift framework to the space-time domain in order to filter video sequences. Nowadays, spatiotemporal filtering techniques can be divided in two types of approaches. Causal techniques only use the past information while omniscient techniques use both past and future information to process data. Causal approaches using mean-shift mode propagation between two consecutive frames have been proposed by [6, 7]. In [7] the method achieves near real-time performance. However, these two methods are not robust to total data occlusion as they only take into account the past frame. Once the modes stop to propagate due to the discontinuity introduced by the occlusion, they cannot be linked to the same data reappearing after the occlusion. In contrast, omniscient techniques consider data as spatiotemporal stacks [8] or as time series [9]. Isotropic extension of standard mean-shift filtering algorithm to spatiotemporal mean-shift has been proposed by [10], with the temporal dimension processed separately from the spatial components. Anisotropic kernel mean-shift was proposed by [11], considering time as a third spatial dimension. The kernels are locally rescaled along the stretch directions in the spatiotemporal stack through an eigenvalues analysis. This has the advantage to adapt kernel orientations to the different structure shapes in the spatiotemporal volume, and to be more robust to initial scale parameters selection. However, robustness to object occlusions is not discussed with this technique.

Later on, in [12] a segmentation method based on hierarchical graph analysis showed better spatiotemporal coherence than mean-shift ones applied to video segmentation but it did not deal with object occlusions. In [9], the approach is only omniscient and can not deal easily with moving objects.

A new spatiotemporal mean-shift formulation that unifies causal and omniscient techniques by a simple adjustment of two temporal scale parameters is introduced in section 2. The proposed clustering algorithm, which merges the samples during the spatiotemporal mean-shift procedure and groups the similar clusters after convergence is detailed in section 3. Results obtained on occluded real video sequence and a qualitative evaluation of the spatiotemporal mean-shift clustering with a real video sequence are presented in section 4.

2. SPATIOTEMPORAL MEAN-SHIFT

In this section, we describe our first contribution: a spatiotemporal filtering approach based on the M-S framework, which can deal with total data occlusion. It can embed within a single formulation causal [6, 7] as well as omniscient [10, 9] techniques.

Let us now consider a set of n samples located at the positions $\{\mathbf{x}_{\mathrm{s},i}\}_{i=1\dots n}$, a set of feature values $\{\mathbf{x}_{\mathrm{r},i}\}_{i=1\dots n}$ and a set of scalar values $\{x_{\mathrm{t},i}\}_{i=1\dots n}$ representing time. The sizes of the spatial and the range dimensions are noted $\mathcal{S}$ and $\mathcal{R}$, respectively. The input data set $\mathbf{X} = \{\mathbf{x}_i\}_{i=1\dots n}$ is defined as

Thanks to CNRS grant PEPS INS2I for funding. This work was performed within the framework of the LABEX PRIMES (ANR-11-LABX-0063) of Université de Lyon, within the program "Investissements d'Avenir" (ANR-11-IDEX-0007) operated by the French National Research Agency (ANR).

follows:

$$\mathbf{x}_i = \left[\mathbf{x}_{\mathrm{s},i}^{'} \, \mathbf{x}_{\mathrm{r},i}^{'} \, x_{\mathrm{t,\,i}} \right]^{'} \quad \text{with} \quad \begin{array}{l} \mathbf{x}_{\mathrm{s},i} \in \mathbb{R}^{\mathcal{S}}, \mathbf{x}_{\mathrm{r},i} \in \mathbb{R}^{\mathcal{R}}, x_{\mathrm{t},i} \in \mathbb{R} \\ i = 1, \ldots, n\text{: samples} \end{array} \tag{1}$$

Considering these notations, we propose the following equation to iteratively compute the spatiotemporal M-S filtering of each sample $\mathbf{x}_i^{[\mathrm{k+1}]}$:

$$\mathbf{x}_i^{[\mathrm{k+1}]} = \frac{\sum_{j=1}^{n} S_{i,j}(\cdot) \, . R_{i,j}(\cdot) \, . T_{i,j}(\cdot) \, . \mathbf{x}_j^{[\mathrm{k}]}}{\sum_{j=1}^{n} S_{i,j}(\cdot) \, . R_{i,j}(\cdot) \, . T_{i,j}(\cdot)} \tag{2}$$

where $S_{i,j}(\cdot)$, $R_{i,j}(\cdot)$ and $T_{i,j}(\cdot)$ are respectively the weighted distances of the spatial, range and temporal domains between a sample of interest $\mathbf{x}_i$ and another sample $\mathbf{x}_j$ ($\mathbf{x}_i$ and $\mathbf{x}_j \in \mathbb{R}^{\mathcal{S}+\mathcal{R}+1}$):

$$S_{i,j}\left(\mathbf{x}_{\mathrm{s},i}^{[\mathrm{k}]}, \mathbf{x}_{\mathrm{s},j}^{[\mathrm{k}]}, \mathbf{H}_{\mathrm{s}} \right) = g_{\mathrm{s}}\Big(d_{\mathrm{s}}^2 \big(\mathbf{x}_{\mathrm{s},i}^{[\mathrm{k}]}, \mathbf{x}_{\mathrm{s},j}^{[\mathrm{k}]}, \mathbf{H}_{\mathrm{s}} \big) \Big) \tag{3}$$

$$R_{i,j}\left(\mathbf{x}_{\mathrm{t},i}^{[\mathrm{k}]}, \mathbf{x}_{\mathrm{t},j}^{[\mathrm{k}]}, \mathbf{H}_{\mathrm{r}} \right) = g_{\mathrm{r}}\Big(d_{\mathrm{r}}^2 \big(\mathbf{x}_{\mathrm{t},i}^{[\mathrm{k}]}, \mathbf{x}_{\mathrm{t},j}^{[\mathrm{k}]}, \mathbf{H}_{\mathrm{r}} \big) \Big) \tag{4}$$

$$T_{i,j}\left(x_{\mathrm{t},i}^{[\mathrm{k}]}, x_{\mathrm{t},j}^{[\mathrm{k}]}, h_{\mathrm{t}^-}, h_{\mathrm{t}^+} \right) = G_{\mathrm{t}}\Big(\varepsilon_{\mathrm{t}} \big(x_{\mathrm{t},i}^{[\mathrm{k}]}, x_{\mathrm{t},j}^{[\mathrm{k}]} \big), h_{\mathrm{t}^-}, h_{\mathrm{t}^+} \Big) \tag{5}$$

In these equations, $d_{\mathrm{s}}(\mathbf{u}_{\mathrm{s}}, \mathbf{v}_{\mathrm{s}}, \mathbf{H}_{\mathrm{s}})$ and $d_{\mathrm{r}}(\mathbf{u}_{\mathrm{r}}, \mathbf{v}_{\mathrm{r}}, \mathbf{H}_{\mathrm{r}})$ are the Mahalanobis distances computed on the spatial and the range domains of two samples $\mathbf{u}$ and $\mathbf{v}$. $\mathbf{H}_{\mathrm{s}}$ and $\mathbf{H}_{\mathrm{r}}$ are respectively the spatial and the range bandwidth matrices of sizes $\mathcal{S} \times \mathcal{S}$ and $\mathcal{R} \times \mathcal{R}$. The Mahalanobis distance is defined by:

$$d(\mathbf{u}, \mathbf{v}, \mathbf{H}) = \left((\mathbf{u} - \mathbf{v})^{'} \, \mathbf{H}^{-1} \, (\mathbf{u} - \mathbf{v}) \right)^{1/2} \tag{6}$$

with $\mathbf{H}$ the bandwidth matrix, squared and positive definite.
In the temporal domain, the difference between a sample of interest $\mathbf{u}$ and a candidate sample $\mathbf{v}$ is computed to take in consideration their temporal order:

$$\varepsilon_{\mathrm{t}}(u_{\mathrm{t}}, v_{\mathrm{t}}) = (v_{\mathrm{t}} - u_{\mathrm{t}}) \tag{7}$$

Then the weights associated to each sample in the spatiotemporal M-S computation will be equal to the combination of the weighted spatial, range and temporal distances. In this work, we use the same profile function g to weight both the spatial and the range distances:

$$g_{\mathrm{s}}\big(d_{\mathrm{s}}^2(\cdot)\big) = g_{\mathrm{r}}\big(d_{\mathrm{r}}^2(\cdot)\big) = \begin{cases} 1 & \text{if } d_{\mathrm{s}}^2(\cdot), \, d_{\mathrm{r}}^2(\cdot) \leq 1 \\ 0 & \text{otherwise} \end{cases} \tag{8}$$

The temporal weighting function is described as follows:

$$G_{\mathrm{t}}\big(\varepsilon_{\mathrm{t}}(\cdot)\big) = \begin{cases} 1 & \text{if } h_{\mathrm{t}^-} \leq \varepsilon_{\mathrm{t}}(\cdot) \leq h_{\mathrm{t}^+} \\ 0 & \text{otherwise} \end{cases} \tag{9}$$

This way setting $h_{\mathrm{t}^+} = 0$ performs causal filtering while setting $\{h_{\mathrm{t}^-} = -\infty, h_{\mathrm{t}^+} = +\infty\}$ performs omniscient filtering. Nevertheless, one can tune h_{t^-} and h_{t^+} in order to adjust the quantity of past and future information to consider, with potentially different ratios.
The proposed formulation of the spatiotemporal M-S filtering given in (2) is a blurring iterative process [1], which ensures convergence in our context [13, 14]. The approach proposed in [6] is a non-blurring M-S.

3. PROPOSED CLUSTERING APPROACH

Data clustering has been developed in two main steps. First step: the clustering algorithm is embedded in the spatiotemporal M-S convergence, which allows us to group samples while they converge and therefore accelerates the procedure over the iterations.
Second step: after projecting the filtered range values in the initial space-time domain, adjacent regions are merged if they show great range and time similarities regarding the scale parameters.

3.1. Coupling clustering and spatiotemporal mean-shift

The first clustering step is based on one assumption: if two samples are close enough in feature space they will converge to the same density maximum. The samples which match this criteria regarding their spatial, range and temporal deviations from each other will be subsequently replaced by their barycenter. Therefore, the number of samples used in the M-S procedure will be reduced together with the computation time. Moreover, a weight is stored for each sample so that each time two samples will be merged their weights are added. Consequently, a sample being the barycenter of ten samples will have ten times more impact than a sample alone. The spatiotemporal M-S filtering of the samples is now defined as:

$$\mathbf{x}_i^{[\mathrm{k+1}]} = \frac{\sum_{j=1}^{n} S_{i,j}(\cdot) \, . R_{i,j}(\cdot) \, . T_{i,j}(\cdot) \, . w_j^{[\mathrm{k}]} . \mathbf{x}_j^{[\mathrm{k}]}}{\sum_{j=1}^{n} S_{i,j}(\cdot) \, . R_{i,j}(\cdot) \, . T_{i,j}(\cdot) \, . w_j^{[\mathrm{k}]}} \tag{10}$$

where $\left\{ w_j^{[\mathrm{k}]} \right\}_{j=1 \ldots n}$ are the sample weights and n is the number of samples at the k-th iteration of the procedure.
Our M-S clustering implementation is detailed in Algorithm 1. The selection of the samples $\mathbf{x}_i$ that can be merged (line:8) is performed using small scale parameters (with respect to the spatial, the range and the temporal scale parameters) and the merging step (line:11) corresponds to their barycenter computation.

3.2. Final merging step

After running M-S, it is common to see samples belonging to homogeneous regions converging towards different density

Algorithm 1 Filtering and clustering algorithm: blurring spatiotemporal M-S with samples merging

Require: $\mathbf{X} = \{\mathbf{x}_i\}_{i=1\ldots n^{[0]}}$
Require: $\mathbf{W} = \{\mathbf{w}_i\}_{i=1\ldots n^{[0]}}$
$\mathrm{k} \leftarrow 0$, $d \leftarrow +\infty$
$\mathbf{X}^{[0]} \leftarrow \mathbf{X}$, $\mathbf{W}^{[0]} \leftarrow \mathbf{W}$
repeat
  for all $\mathbf{x}_i \in \mathbf{X}^{[\mathrm{k}]}$ **do**
    Compute $\mathbf{x}_i^{[\mathrm{k}+1]}$ with equation (10)
  end for
  $d \leftarrow |\mathbf{X}^{[\mathrm{k}+1]} - \mathbf{X}^{[\mathrm{k}]}|$
  $\mathbf{M} \leftarrow$ all $\mathbf{x}_i^{[\mathrm{k}]} \in \mathbf{X}^{[\mathrm{k}]}$ that can be merged
  repeat
    Extract one $\mathbf{x}_i$ from $\mathbf{M}$
    $\mathbf{x}_i^{[\mathrm{k}+1]} \leftarrow$ Merge $\mathbf{x}_i^{[\mathrm{k}+1]}$ with its neighbors in $\mathbf{M}$
    $\mathbf{W}^{[\mathrm{k}+1]} \leftarrow$ Update the weight assigned to $\mathbf{x}_i^{[\mathrm{k}+1]}$
    Remove $\mathbf{x}_i^{[\mathrm{k}+1]}$ neighbors entries from $\mathbf{X}^{[\mathrm{k}+1]}$, $\mathbf{W}^{[\mathrm{k}+1]}$ and $\mathbf{M}$
  until $\mathbf{M}$ is empty
  $k \leftarrow k + 1$
until $d < \epsilon$: Stopping criteria equal to 0.01
$\hat{\mathbf{X}} \leftarrow \mathbf{X}^{[\mathrm{k}+1]}$
return $\hat{\mathbf{X}}$

Algorithm 2 Final merging algorithm

Require: $\mathbf{X} = \{\mathbf{x}_i\}_{i=1\ldots n^{[0]}}$
Require: $\hat{\mathbf{X}} = \{\hat{\mathbf{x}}_i\}_{i=1\ldots n^{[\mathrm{k}]}}$: Output of algorithm 1
repeat
  $\hat{\mathbf{X}}_{\mathrm{tmp}} \leftarrow \hat{\mathbf{X}}$
  for all $\hat{\mathbf{x}}_i \in \hat{\mathbf{X}}$ **do**
    $\mathcal{N} \leftarrow$ Find $\hat{\mathbf{x}}_i$ neighbors compared to $\mathbf{H}_{\mathrm{r}}$
    $\mathcal{S} \leftarrow$ Propagate the clusters of $\mathcal{N}$ and $\hat{\mathbf{x}}_i$ in the initial sequence and extract the clusters which regions are connected to the region associated to $\hat{\mathbf{x}}_i$
    $\mathcal{T} \leftarrow$ Propagate the clusters of $\mathcal{S}$ and $\hat{\mathbf{x}}_i$ in the initial sequence and extract the clusters which temporal deviation between its initial samples and the initial samples of $\hat{\mathbf{x}}_i$ lies between h_{t^-} and h_{t^+}
    $\hat{\mathbf{x}}_i \leftarrow$ Merge $\hat{\mathbf{x}}_i$ with $\mathcal{T}$ clusters
    $\hat{\mathbf{X}} \leftarrow$ Remove $\mathcal{T}$ clusters from $\hat{\mathbf{X}}$
  end for
until Size($\hat{\mathbf{X}}$) = Size($\hat{\mathbf{X}}_{\mathrm{tmp}}$)
return $\hat{\mathbf{X}}$

maximums. This happens when the spatial scale used is too low compared to the sizes of these regions. To overcome this problem we propose the merging algorithm detailed in Algorithm 2: the clusters which inter-class range distances $d_{\mathrm{r}}(\cdot, \cdot, \mathbf{H}_{\mathrm{r}})$ are inferior to one, after convergence, are candidates for merging. The assumption is that without considering the spatial and the temporal domains, close range values should be merged. However samples with close range values might not be part of the same object nor appear at the same time in the video sequence. Thus, the second step consists in checking in the initial sequence if the samples of the candidate clusters belong to connected regions and if the minimum temporal distance between their samples is included between h_{t^-} and h_{t^+}. Finally, the clusters returned by the spatiotemporal M-S procedure will be merged if they respect these three conditions. As in the clustering part, the merging consists in computing the barycenter of several clusters.

4. EXPERIMENTS AND RESULTS

In this section, we present the results obtained on two real datasets chosen to highlight object occlusion management.
In our experiments, we have empirically chosen to merge two samples if their spatial, range and temporal distances are all at least thirty time inferior to their scale parameter. This has shown to be a good criteria to accelerate the computation time without degrading the filtering results.

4.1. Table Tennis sequence

The method was tested on the *Table Tennis* video available on the internet and also used by [6]. In order to test the proposed procedures, five frames were extracted and an occlusion was added on all frames so that the ball is completely masked at frame 30 (Fig.1).
Before processing the frames, their RGB characteristics were transformed into the Lab space to improve M-S efficiency [15]. We manually determined spatial and range bandwidth parameters ($\mathbf{H}_{\mathrm{s}}$ and $\mathbf{H}_{\mathrm{r}}$ respectively) after studying size of objects and range values of each Lab component. The same matrices were used for all experiments. We focus on the tuning the temporal scales h_{t^-} and h_{t^+} in order to obtained comparable results than ones obtained by [6, 7].
Figure 1 shows the clustering outputs obtained for three h_{t^-} and h_{t^+} pairs, allowing both causal and omniscient filtering. The class labels are represented in color for convenience.
In all cases, we can see that the objects are always well segmented. However, the ball is lost after its occlusion by the rectangle when considering only the past frame and a new classes are created at frame 31. In contrast, when considering 3 past frames the proposed method manages to find a similar cluster in the past for the ball and will always assign the same label to it. Such result confirms the ability of the clustering to deal with occlusions without creating a new class for an object that has temporarily disappeared. Using larger temporal scales reduce artifact regions as the ball's shadow and the contours of the ping pong bat. One can note that a median filter (or any small regions removal scheme) can be applied on the clustering results in order to improve clusters homogeneity.

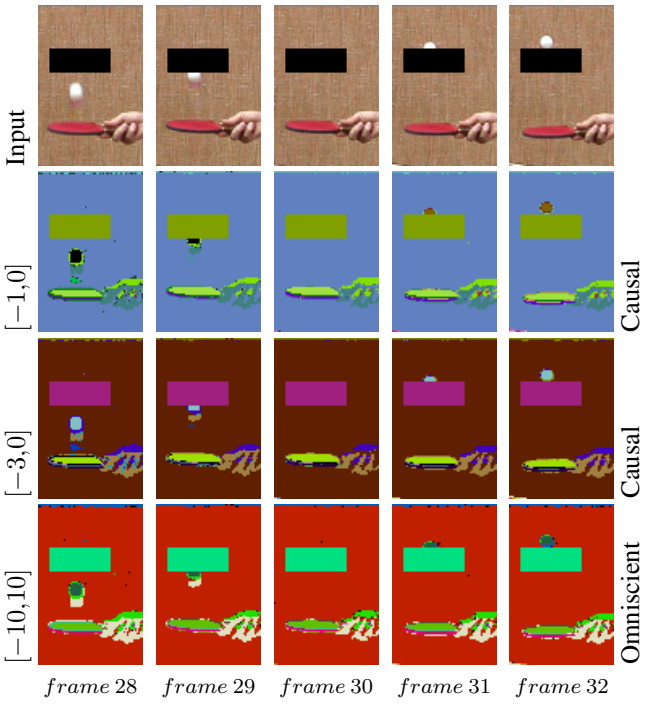

Fig. 1. Table Tennis sequence results. The first line contains the extracted frames. The remaining lines are clustering results obtained using the temporal scales specified at each line beginning. All experiments are performed with $\mathbf{H}_s = diag(40, 40)$ and $\mathbf{H}_r = diag(50, 30, 30)$. No small regions removal was used, raw clustering output.

4.2. Soccer sequence

The method was tested on a soccer video freely available on the internet[1]. Four frames were extracted over one second of capture so that the soccer jersey of one player is completely masked by the other at some point (Fig.2). This sequence was pre-processed likewise the Table Tennis sequence.

Figure 2 shows the outputs obtained by our method on real data. While the filtered frames show coherent clustering results compared to the original data, some regions as the playground or the player pants are over segmented and the number five is lost at time 3 when considering only the past frame. Moreover the class of the red soccer jersey changes after its occlusion at frame 3. In contrast, when considering three frames in the past there is an improvement of the clustering of the playground, the player shadows, their pants and of the snow in the background. The temporal coherence of the clusters is preserved, relatively to the scales chosen, and the same class is always assigned to the red soccer jersey.

Nevertheless, we can see the class of the number five changing between the last two frames. This comes from a "space-time trade-off" that has to be done between choosing a high spatial scale to compensate an hypothetical high displacement of an object during its occlusion and risking to link classes that are similar in range and time but that are too close regarding the spatial scale. We chose a high spatial scale to be insensitive to the red player displacement but the number five was clustered with the background when it becomes too close. Thus, dealing properly with total occlusions requires a fine tuning of the spatial scale.

[1]Soccer sequence: `ftp://ftp.tnt.uni-hannover.de/pub/svc/testsequences/`

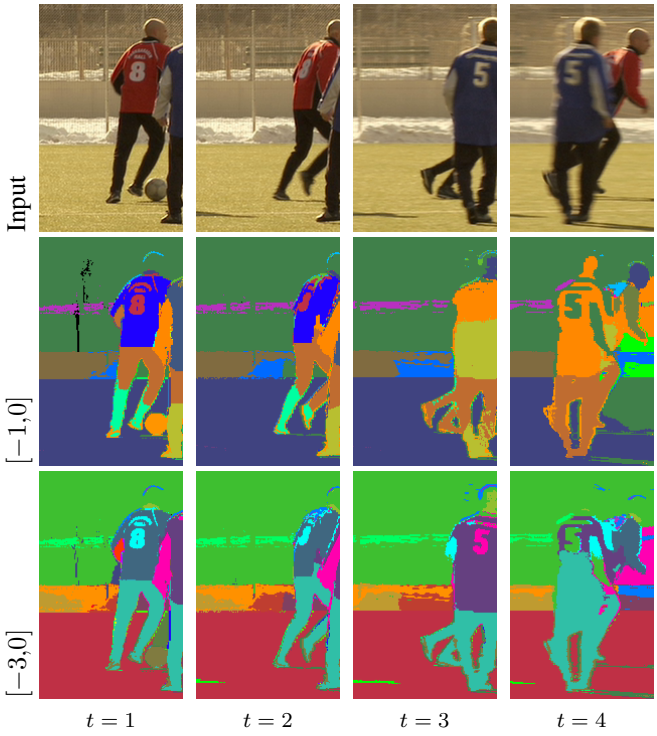

Fig. 2. Soccer sequence results. The first line contains the extracted frame. The second line is the result of our method with $\mathbf{H}_s = diag(20, 20)$, $\mathbf{H}_r = diag(10, 10, 10)$, $h_{t^-} = -1, h_{t^+} = 0$. The third line is the result of our method with $\mathbf{H}_s = diag(20, 20)$, $\mathbf{H}_r = diag(10, 10, 10)$, $h_{t^-} = -3, h_{t^+} = 0$.

5. CONCLUSION

We have introduced a new spatiotemporal mean-shift formulation, general enough to describe behaviors of many existing approaches, able to cluster data in a causal or omniscient way by allowing to set independently the amount of past and future information to consider. We then proposed a new clustering procedure embedded in the mean-shift process that allows to merge samples during the convergence process and therefore accelerates the computation time. The ability of our method to preserve the temporal coherence of the clusters after total occlusions has been shown on real data.

6. REFERENCES

[1] K. Fukunaga and L. D. Hostetler, "Estimation of the gradient of a density function with applications in pattern recognition.," *Information Theory, IEEE Transaction on*, vol. 21, no. 1, pp. 32–40, 1975.

[2] D. Comaniciu and P. Meer, "Mean shift: A robust approach toward feature space analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 5, pp. 603–619, 2002.

[3] D. Comaniciu, V. Ramesh, and P. Meer, "Kernel-based object tracking," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 25, no. 5, pp. 564–577, May 2003.

[4] V. Bruni and D. Vitulano, "An improvement of kernel-based object tracking based on human perception," *Systems, Man, and Cybernetics: Systems, IEEE Transactions on*, vol. 44, no. 11, pp. 1474–1485, Nov 2014.

[5] Wei Feng and Rong-Chun Zhao, "Non-rigid objects detection and segmentation in video sequence using 3d mean shift analysis," in *International Conference on Machine Learning and Cybernetics*, Nov 2003, vol. 5, pp. 3134–3139 Vol.5.

[6] IreneY.H. Gu, Vasile Gui, and Zhifei Xu, "Video segmentation using joint space-time-range adaptive mean shift," in *Advances in Multimedia Information Processing - PCM 2006*, Yueting Zhuang, Shi-Qiang Yang, Yong Rui, and Qinming He, Eds., vol. 4261 of *Lecture Notes in Computer Science*, pp. 740–748. Springer Berlin Heidelberg, 2006.

[7] Sylvain Paris, "Edge-preserving smoothing and mean-shift segmentation of video streams," in *Computer Vision – ECCV 2008*, David Forsyth, Philip Torr, and Andrew Zisserman, Eds., vol. 5303 of *Lecture Notes in Computer Science*, pp. 460–473. Springer Berlin Heidelberg, 2008.

[8] Daniel DeMenthon and David Doermann, "Video retrieval using spatio-temporal descriptors," in *Proceedings of the eleventh ACM international conference on Multimedia*. ACM, 2003, pp. 508–517.

[9] Simon Mure, Thomas Grenier, Dominik S. Meier, Charles R.G. Guttmann, and Hugues Benoit-Cattin, "Unsupervised spatio-temporal filtering of image sequences. a mean-shift specification," *Pattern Recognition Letters*, vol. 68, Part 1, pp. 48 – 55, 2015.

[10] Tomoyuki Nagahashi, Hironobu Fujiyoshi, and Takeo Kanade, "Video segmentation using iterated graph cuts based on spatio-temporal volumes," in *Computer Vision – ACCV 2009*, Hongbin Zha, Rin-ichiro Taniguchi, and Stephen Maybank, Eds., vol. 5995 of *Lecture Notes in Computer Science*, pp. 655–666. Springer Berlin Heidelberg, 2010.

[11] Jue Wang, Bo Thiesson, Yingqing Xu, and Michael Cohen, "Image and video segmentation by anisotropic kernel mean shift," in *Computer Vision - ECCV 2004*, Tomás Pajdla and Jiří Matas, Eds., vol. 3022 of *Lecture Notes in Computer Science*, pp. 238–249. Springer Berlin Heidelberg, 2004.

[12] M. Grundmann, V. Kwatra, Mei Han, and I. Essa, "Efficient hierarchical graph-based video segmentation," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, June 2010, pp. 2141–2148.

[13] Y. Cheng, "Mean shift, mode seeking, and clustering," *Pattern Analysis and Machine Intelligence, IEEE Transaction on*, vol. 17, no. 8, pp. 790–799, 1995.

[14] Mark Fashing and Carlo Tomasi, "Mean shift is a bound optimization," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 27, no. 3, pp. 471–474, 2005.

[15] Ting Li, Thomas Grenier, and Hugues Benoit-Cattin, "Color space influence on mean shift filtering," in *Image Processing (ICIP), 2011 18th IEEE International Conference on*. IEEE, 2011, pp. 1469–1472.