On the Contribution of Saliency in Visual Tracking

Iman Alikhani[1], Hamed R.-Tavakoli[2], Esa Rahtu[3] and Jorma Laaksonen[2]
[1]*Department of Computer Science and Engineering, University of Oulu, Oulu, Finland*
[2]*Department of Computer Science, Aalto University, Espoo, Finland*
[3]*Center for Machine Vision Research, University of Oulu, Oulu, Finland*

Keywords: Saliency, Mean-shift Tracking, Target Representation.

Abstract: Visual target tracking is a long-standing problem in the domain of computer vision. There are numerous methods proposed over several years. A recent trend in visual tracking has been target representation and tracking using saliency models inspired by the attentive mechanism of the human. Motivated to investigate the usefulness of such target representation scheme, we study several target representation techniques for mean-shift tracking framework, where the feature space can include color, texture, saliency, and gradient orientation information. In particular, we study the usefulness of the joint distribution of color-texture, color-saliency, and color-orientation in comparison with the color distribution. The performance is evaluated using the *visual object tracking* (VOT) 2013 which provides a systematic mechanism and a database for the assessment of tracking algorithms. We summarize the results in terms of accuracy & robustness; and discuss the usefulness of saliency-based target tracking.

1 INTRODUCTION

Visual object tracking is an old problem in computer vision with application in surveillance, traffic control, object-based video compression, video indexing, human-computer interaction, traffic monitoring, and etc. The literature of computer society is full of various techniques such as point tracking based methods, kernel tracking schemes and silhouette tracking approaches (Yilmaz et al., 2006). A point tracker relies on points in order to associate an object with its state in previous frames. Silhouette-based approaches take advantage of object region estimation in each frame where the object region is often represented in terms of a shape model (e.g., contours) and is tracked over time by a matching mechanism. Kernel-based techniques encode the object shape and appearance as a template with an associated density, named kernel, and track the object in successive frames.

The tracking algorithms rely on a means of target representation. The target representation is influenced by both algorithm and constraints such as computational power, necessity of training free start, robustness, illumination change, and etc. In many interactive scenarios, with real-time requirements, feature based target representation is desirable where simple features like color cues are preferred. Recently, a trend in such cases have been the use of saliency-based features in order to boost the tracking performance. In this context, the models get the advantage of saliency in target representation, e.g., (Frintrop and Kessel, 2009) proposed a cognitive observation model inspired by human visual system. (Borji et al., 2012) extended the model of (Frintrop and Kessel, 2009) to incorporate the background context in a CONDENSATION-based tracker. The saliency-based target representation was also adopted by (Tavakoli and Moin, 2010; Plataniotis and Venetsanopoulos, 2000) in a kernel-based framework (Comaniciu et al., 2003), where the joint distribution of color and saliency is utilized. These methods show effective contribution of salience, albeit on a limited number of sequences or particular task specific videos. This provides the motivation for investigating saliency contribution on a fairly general database in order to identify its strength and weaknesses for object tracking.

We specifically address the following question *"How well does mean-shift algorithm benefit from saliency-based target representation?"*. To find an answer, we explore various features useful for target representations and assess the performance of object tracking using them. We build target models using joint distribution of color with edge, texture,

Alikhani, I., R.-Tavakoli, H., Rahtu, E. and Laaksonen, J.
On the Contribution of Saliency in Visual Tracking.
DOI: 10.5220/0005648900170021
In *Proceedings of the 11th Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2016) - Volume 4: VISAPP*, pages 17-21
ISBN: 978-989-758-175-5

and saliency. Afterwards, the performance of feature combinations are evaluated in different circumstances within a mean-shift tracking framework.

2 TRACKING FRAMEWORK

The mean-shift tracking is an ideal framework for the analysis of various target representations because as a deterministic framework, it guarantees insensitive to algorithm results. The initial idea of mean-shift was introduced in (Fukunaga and Hostetler, 1975) and developed by (Comaniciu et al., 2003) into an efficient tracking procedure, which is still useful in many real-time applications. It requires a reference model to represent the target which is achieved by using the probability distribution function (PDF) of features within a target patch. Having a normalized target patch, $\{x_i^*\}_{i=1,..,n}$ consisting of n pixels, the target model is defined as $q = \{q_f\}_{f=1...m}$ where

$$\{q_f\}_{f=1,2,..,m} = \alpha \sum_{i=1}^{n} \mathcal{K}\left(\|x_i^*\|^2\right) \delta\left(\mathrm{b}(x_i^*) - f\right), \quad (1)$$

$\{q_f\}$ is the probability of feature f in target model q, $\alpha = 1/\sum_{i=1}^{n} \mathcal{K}\left(\|x_i^*\|^2\right)$ is a normalizing constant, $\mathcal{K}(.)$ is an isotropic kernel, δ is the Kronecker delta function and $\mathrm{b} : \mathbb{R}^2 \rightarrow \{1 \ldots m\}$ is a mapping function to associate a pixel to its corresponding bin in the feature space PDF.

Once the tracking framework has a target model, it starts the tracking procedure by examining target candidates. A target candidate, $p = \{p_f\}_{f=1...m}$, is obtained by examining a candidate patch $\{x_i\}_{i=1,..,n}$ centered at location y where

$$\{p_f\}_{f=1,2,..,m} = \beta_h \sum_{i=1}^{n} \mathcal{K}\left(\|(y - x_i)/h\|^2\right) \delta\left(\mathrm{b}(x_i) - f\right), \quad (2)$$

$\{p_f\}$ is the probability of feature f in candidate model p, $\beta_h = 1/\sum_{i=1}^{n} \mathcal{K}\left(\|(y - x_i)/h\|^2\right)$ is a normalizing constant, and $\mathcal{K}(.)$ is an isotropic kernel with bandwidth h. The tracking procedure tries to maximize the Bhattacharyya coefficient between p and q, $bc(q, p_y) = \sum_f \sqrt{p_{f,y} q_f}$, to obtain the new location of the target, denoted as y_{new}. The procedure can be formulated as an iterative distance minimization problem (Comaniciu et al., 2003) which is summarized as follows:

$$y_{new} = \frac{\sum_i x_i w_i}{\sum_i w_i}, \quad (3)$$

$$w_i = \sum_f \sqrt{\frac{q_f}{p_{f,y}}} \delta(\mathrm{b}(x_i^*) - f), \quad (4)$$

where $p_{f,y}$ indicates a candidate model located at y. Thus, the complete tracking procedure can be summarized in Algorithm 1.

Algorithm 1: Mean-Shift Algorithm.

Require: target model q, location of target (y) from previous frame, minimum distance ε, number of maximum iteration N

1: Initialize number of iterations, $t \leftarrow 0$.
2: **repeat**
3: make a candidate model p.
4: compute y_{new} using (3)
5: $d \leftarrow \|y_{new} - y\|$
6: $y \leftarrow y_{new}$ and $t \leftarrow t + 1$
7: **until** $t < N$ and $\varepsilon < d$

3 MODEL REPRESENTATION

The mean-shift framework relies on the probability density function of features in order to model and track a target. The original mean-shift algorithm utilizes color space as a global feature. The color space is often a well discriminative feature and is an easily accessible piece of information. The distribution of color is encoded in terms of a histogram of quantized colors. We adopt the RGB feature representation as a baseline and evaluate the other target models against it. To model the target using RGB, we quantize each color channel into 8 bins and form a histogram by concatenating the histogram of each channel resulting in a feature space of size $512(8 \times 8 \times 8)$. We identify this model representation as 'RGB'.

The RGB-based mean-shift tracking is prone to missing target when the appearance of target and background are similar. In order to improve the robustness and prevent target loss, Ning *et al.* (Ning et al., 2009) incorporated the texture information. They utilized texture information in terms of a modified local binary patterns (LBP) (Ojala et al., 2002) to efficiently extract edge and corner like areas as textured regions and build a joint distribution of color and texture. We adopt the same approach to model target using color-texture which results in a feature space of 2560. The same parameter setting is utilized; therefore each color channel is quantized into 8 bins and a modified LBP operator of 5 bins is used[1]. Furthermore, we also investigate target modeling using the original LBP operator. We refer to the first target model, proposed by (Ning et al., 2009), as 'RGB-LBP5' and call the latter model 'RGB-LBP'.

[1]We use the source code provided by authors which is available online at: http://www4.comp.polyu.edu.hk/~cslzhang/code/IJPRAI_demo_software.zip

Saliency-based target representation has been favored to replace the LBP-based target representation in (Tavakoli and Moin, 2010). The saliency is measured using a similarity operator, called local similarity number (LSN) (Tavakoli et al., 2013). It is defined in terms of the amount of similarity between a pixel and its surrounding pixels, where a salient pixel is less similar to its surroundings. The saliency values can be quantized to define the amount of pop-out quality of a pixel. We use the same settings as (Tavakoli et al., 2013) where the saliency values corresponding to the low saliency regions are suppressed to zero and discarded. Then, the joint distribution of the color-saliency is built using color and the remaining saliency values quantized into 5 bins. By suppressing the low saliency regions to zero, (Tavakoli et al., 2013) preserves the unity of the target model and neglect the non-salient regions within the target area resulting in target miss in some circumstances. Thus, we also study an alternative model in which the low saliency values are not suppressed and the whole saliency value span is quantized into 9 bins, i.e., we use

$$\mathrm{LSN}^{d}_{8,1} = 1 + \sum_{i=0}^{7} \hat{f}(g_i - g_c, d), \tag{5}$$

$$\hat{f}(x,d) = \begin{cases} 1 & |x| \leq d \\ 0 & |x| > d \end{cases}, \tag{6}$$

where g_c and g_i are the gray scale values at the center and i-th neighboring pixel and d is a similarity margin. These target models are identified as 'RGB-LSN5' and 'RGB-LSN', respectively.

The concept of saliency modeling in target representation is putting weight on the visually attractive regions which can be modeled using edges. In fact, we also study normalized gradient orientation for target modeling. The normalized gradient orientation is a discriminative feature which is resistant towards noise and illumination change and is applied in many visual descriptors such as (Dalal and Triggs, 2005). To utilize gradient orientation, we convolve the image patch with $F_x = [1,0,-1]$ and $F_y = [1,0,-1]^T$ and quantize the orientation into 9 bins. This feature encodes the edginess of the target. Afterwards, the joint distribution of color and gradient orientation is built. This results in a feature space of size $8 \times 8 \times 8 \times 9$, 8 bins per color channel and 9 bins for the gradient orientation. We call this target model 'RGB-OG' in the rest of the paper.

4 EXPERIMENTS

We use the *visual object tracking* (VOT) 2013 toolkit and benchmark (Kristan et al., 2013). It consists of various real-life sequences proved by a rich and small corpus of 16 videos. Since each frame of a video is labeled with a visual attribute to reflect a particular challenge: (i) occlusion, (ii) illumination change, (iii) motion change, (iv) size/scale change, (v) camera motion, and (vi) non-degraded, adopting VOT helps providing an insight about the behavior of each target representation in encountering these circumstances. The performance metrics provided by the VOT toolkit are *accuracy* and *robustness*. The accuracy is defined in terms of the overlap between the tracker predicted bounding box and the ground-truth bounding box. The robustness is measured by the failure rate obtained from counting the number of times the tracker drifted from the target, where the overlap between prediction and ground-truth is zero. All the experiments are repeated several times, and the average repetitions are reported. (Please consult VOT documents for the details (Kristan et al., 2013).) To validate the role of target representation in convergence speed, we complemented the VOT metrics with an average convergence metric. It is estimated by averaging the number of iterations it takes for the algorithm to converge into a stable candidate model for all the valid frames (the frames where the target is not lost).

Table 1: Average iteration until convergence. The average iteration is reported for all the valid frames of the baseline experiment.

Target Model	Average Iteration
RGB-LSN	4.32
RGB-LBP	4.38
RGB-OG	4.38
RGB	4.38
RGB-LSN5 (Tavakoli et al., 2013)	4.67
RGB-LBP5 (Ning et al., 2009)	4.96

Three experiments are performed: baseline, initialization perturbation and gray-scale. The baseline experiment tests the tracker performance against all the sequences. The region noise assesses the performance by perturbing the bounding-boxes at the initialization, and gray-scale experiment repeats the baseline experiment by converting the sequences into gray-scale. In order to perform the gray-scale experiment, we treated the gray-scale frames as color by replicating the gray channel into three channels. To assess the convergence speed, we re-implemented the baseline experiment to report the convergence rate of methods for valid frames.

Table 2 summarizes the overall ranking results of the evaluated target models. We performed the ranking of algorithms in conjunction with the 2013 chal-

Table 2: Ranking results of the target models, The per-accuracy and per-robustness averaged ranks are denoted as Acc and Rub, respectively. The average ranking per experiment is denoted as Avg. and the average column contains accuracy and robustness over all the experiments followed and the total rank column reports the overall average performance in regard to accuracy and robustness. Red indicates the best performance, green represents the second best and the blue highlights the third best.

Target Model	Baseline			Initialization perturbation			Gray-scale			Average		Total
	Acc	Rub	Avg.	Acc	Rub	Avg.	Acc	Rub	Avg.	Acc	Rub	Rank
RGB-LSN	18.71	16.51	17.6	16.31	17.84	17.08	20.17	18.07	19.12	18.39	17.48	17.93
RGB-LBP	17.32	20.24	18.78	16.49	17.10	16.79	18.58	17.98	18.28	17.47	18.44	17.95
RGB-OG	17.08	20.05	18.56	16.97	17.97	17.47	18.01	19.23	18.62	17.36	19.08	18.22
RGB	19.05	16.43	17.74	16.13	16.70	16.41	22.39	23.26	22.83	19.19	18.80	18.99
RGB-LBP5 (Ning et al., 2009)	22.99	19.49	21.24	20.23	18.68	19.46	21.94	23.75	22.85	21.72	20.64	21.18
RGB-LSN5 (Tavakoli et al., 2013)	30.25	21.50	25.88	27.99	20.98	24.49	26.33	28.16	27.25	28.19	23.54	25.87

Table 3: Detailed performance of the models in each visual attribute in terms of overlap (accuracy) and failures (robustness). The rank indicates the average ranking considering both accuracy rank and robustness rank.

	Target Model	Camera motion			Illumination change			Occlusion			Size/scale change			Motion change			non-degraded		
		Over	Fail	Rank	Over	Fail	Rank	Over	Fail	Rank	Over	Fail	Rank	Over	Fail	Rank	Over	Fail	Rank
Baseline	RGB-LSN	0.55	18.00	19.32	0.39	5.00	28.28	0.58	2.00	15.87	0.43	1.00	11.66	0.57	13.00	18.59	0.68	0.00	11.95
	RGB-LBP	0.55	22.00	21.82	0.39	3.00	25.31	0.60	4.00	18.26	0.45	4.00	14.89	0.57	12.00	20.46	0.68	0.00	11.95
	RGB-OG	0.56	21.00	18.98	0.41	3.00	24.02	0.61	3.00	14.11	0.45	7.00	18.13	0.56	14.00	21.5	0.69	0.00	12.00
	RGB	0.55	16.00	18.43	0.39	4.00	25.25	0.59	2.00	14.62	0.42	2.00	14.79	0.55	10.00	19.91	0.67	0.00	13.00
	RGB-LSN5 (Tavakoli et al., 2013)	0.50	16.00	22.72	0.36	7.00	28.00	0.35	7.00	28.19	0.38	8.00	19.00	0.47	26.00	25.59	0.33	2.00	31.75
	RGB-LBP5 (Ning et al., 2009)	0.55	13.00	17.6	0.41	4.00	25.75	0.51	3.00	21.86	0.41	6.00	13.7	0.55	16.00	20.54	0.44	1.00	27.96
Init Perturb	RGB-LSN	0.53	19.07	17.57	0.38	4.47	26.36	0.55	2.67	16.56	0.42	4.53	14.25	0.53	14.07	17.82	0.65	0.00	9.92
	RGB-LBP	0.52	20.60	18.67	0.39	3.80	25.75	0.55	3.20	16.75	0.43	3.00	10.92	0.53	12.40	17.79	0.64	0.00	10.90
	RGB-OG	0.52	19.47	19.99	0.39	2.80	24.00	0.57	2.80	14.31	0.43	6.07	14.70	0.52	12.67	19.35	0.63	0.00	12.50
	RGB	0.52	16.80	17.77	0.38	4.20	26.09	0.57	2.33	13.44	0.43	3.40	13.04	0.52	12.13	18.17	0.64	0.00	10.00
	RGB-LSN5 (Tavakoli et al., 2013)	0.51	18.47	20.54	0.39	7.47	27.88	0.39	5.87	24.00	0.38	8.47	18.75	0.48	27.13	24.73	0.30	2.93	31.00
	RGB-LBP5 (Ning et al., 2009)	0.54	14.33	14.78	0.40	4.33	25.36	0.50	2.60	20.17	0.42	6.07	10.54	0.53	17.07	17.38	0.42	1.60	28.50
Gray-scale	RGB-LSN	0.50	37.00	20.19	0.35	7.00	24.57	0.60	4.00	16.21	0.39	9.00	20.04	0.47	20.00	20.15	0.62	0.00	13.59
	RGB-LBP	0.52	44.00	20.48	0.39	3.80	24.32	0.55	3.20	14.24	0.43	3.00	14.41	0.53	12.40	21.50	0.64	0.00	14.75
	RGB-OG	0.50	46.00	20.47	0.36	7.00	23.57	0.60	3.00	19.42	0.43	10.00	14.45	0.47	24.00	20.92	0.61	0.00	10.92
	RGB	0.47	53.00	25.09	0.35	11.00	27.75	0.55	5.00	17.17	0.39	10.00	20.65	0.45	26.00	24.38	0.60	1.00	21.92
	RGB-LSN5 (Tavakoli et al., 2013)	0.47	50.00	26.38	0.37	11.00	25.47	0.36	21.00	28.29	0.37	23.00	27.34	0.45	46.00	28.00	0.33	3.00	28.00
	RGB-LBP5 (Ning et al., 2009)	0.49	44.00	22.42	0.41	7.00	22.00	0.38	11.00	22.77	0.40	15.00	21.15	0.49	37.00	23.50	0.35	1.00	25.25

lenge algorithms[2], though we only report the models of our interest. (Please check the supplement for the ranking in conjunction with the algorithms in the challenge.) Surprisingly, we learn that the target representation models of RGB-LBP5 (Ning et al., 2009) and RGB-LSN5 (Tavakoli et al., 2013) does not perform better than the original RGB target model in terms of accuracy and robustness. The same conclusion can be inferred in terms of convergence rate as reported in Table 1. We credit the finding to the limited number of test sequences in (Ning et al., 2009; Tavakoli et al., 2013) which could have affected the understanding about the models behavior and performance. It is worth noting that both methods are reported having good performance on tracking small targets on difficult backgrounds; however, a small number of VOT sequences have such a characteristic.

In order to understand the behavior of each target model better, we analyze their ranking performance in the six visual attributes a frame can be associated with. The result is reported in Table 3. There is no single model that performs the best in all categories. In the presence of camera motion, RGB-LBP5 (Ning et al., 2009) performs best during the first two experiments (i.e., baseline and initialization perturbation) by having less failures, but it is replaced by RGB-LSN for the gray-scale experiment. In the illumination change circumstance, the target model using oriented gradient and RGB features outperform other representations in the first two experiments; these features are also the best in the case of occlusions during baseline experiment. The RGB-LBP model is the foremost model in the case of size change in the presence gray-scale input. It is, however, outperformed by RGB-LSN model for the baseline experiment. The RGB-LSN model performs the best in the presence of motion change and non-degraded sequences by having high overlap and low failure rate. The RGB-LSN has an above average performance in all the circumstances, which results in a winning overall ranking verdict in VOT 2013 experiments.

An interesting observation is that the RGB model often has an average or above average performance on most of the sequences during the baseline and initialization perturbation. This indicates that poten-

[2]The VOT2013 results were obtained from: http://box.vicos.si/vot/vot2013_results.zip

tially the RGB model can be enough for many tasks when color information is present. To assess this hypotheses, we reordered the model ranking based on baseline and initialization perturbation results: RGB $(17.08) >$ RGB-LSN $(17.34) >$ RGB-LBP $(17.79) >$ RGB-OG $(18.13) >$ RGB-LBP5 $(20.35) >$ RGB-LSN5 (25.18). It is clearly obvious that dropping the requirement for gray-scale handling an RGB target model is enough to perform better than any other model.

5 CONCLUSION AND DISCUSSION

"How well does mean-shift algorithm benefit from saliency-based target representation?"

Despite we demonstrated that RGB model is sufficient for many cases, the answer is not a straight forward yes/no. In fact, the choice of target model is dependent on the application, target characteristic and sensor. We must however assert that RGB-LSN has slightly the edge over RGB since it also handles the gray-scale input properly. Otherwise putting the gray-scale input aside, we can interpret that the contribution of the combination of various features with RGB is often marginal and most probably not needed.

It may be questioned about the reason why the findings in this study may differ from those papers which proposed the application of various features with RGB for target modeling. The reason most probably lies on the use of limited number of test sequences and target specific (e.g., tracking a particular object) applications which affects the understanding about the general behavior of the model. A similar phenomenon is often observed in the evaluation of saliency-based trackers which casts a shadow on their true strength and motivates a careful study of saliency based algorithms and methods. This issue goes outside the span of the current paper and will be addressed in the future work.

It is also worth noting that relying solely on the average ranking results of a benchmark is not necessarily wise and a closer look to the underlying scores are needed. This becomes important in choosing the appropriate algorithm and target model for a specific application since the overall score can be easily skewed and be misleading.

ACKNOWLEDGEMENTS

Hamed R.-Tavakoli and Jorma Laaksonen were supported by The Academy of Finland under the Finnish Center of Excellence in Computational Inference Research (COIN).

REFERENCES

Borji, A., Frintrop, S., Sihite, D., and Itti, L. (2012). Adaptive object tracking by learning background context. In *CVPRW*, pages 23–30.

Comaniciu, D., Ramesh, V., and Meer, P. (2003). Kernel-based object tracking. *IEEE Trans. Pattern Anal. Mach. Intell.*, 25(5):564–577.

Dalal, N. and Triggs, B. (2005). Histograms of oriented gradients for human detection. In *CVPR*, volume 1, pages 886–893 vol. 1.

Frintrop, S. and Kessel, M. (2009). Most salient region tracking. In *ICRA*.

Fukunaga, K. and Hostetler, L. (1975). The estimation of the gradient of a density function, with applications in pattern recognition. *IEEE Trans. Inf. Theory*, 21(1):32–40.

Kristan, M., Pflugfelder, R., Leonardis, A., Matas, J., Porikli, F., Cehovin, L., Nebehay, G., Fernandez, G., Vojir, T., Gatt, A., Khajenezhad, A., Salahledin, A., Soltani-Farani, A., Zarezade, A., Petrosino, A., Milton, A., Bozorgtabar, B., Li, B., Chan, C. S., Heng, C., Ward, D., Kearney, D., Monekosso, D., Karaimer, H., Rabiee, H., Zhu, J., Gao, J., Xiao, J., Zhang, J., Xing, J., Huang, K., Lebeda, K., Cao, L., Maresca, M., Lim, M. K., El Helw, M., Felsberg, M., Remagnino, P., Bowden, R., Goecke, R., Stolkin, R., Lim, S., Maher, S., Poullot, S., Wong, S., Satoh, S., Chen, W., Hu, W., Zhang, X., Li, Y., and Niu, Z. (2013). The visual object tracking vot2013 challenge results. In *ICCVW*, pages 98–111.

Ning, J., Zhang, L., Zhang, D., and Wu, C. (2009). Robust object tracking using joint color-texture histogram. *International Journal of Pattern Recognition and Artificial Intelligence*, 23(07):1245–1263.

Ojala, T., Pietikäinen, M., and Mäenpää, T. (2002). Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(7):971–987.

Plataniotis, K. N. and Venetsanopoulos, A. N. (2000). *Color Image Processing and Applications*. Springer-Verlag New York, Inc., New York, NY, USA.

Tavakoli, H. and Moin, M. (2010). Mean-shift video tracking using color-lsn histogram. In *Telecommunications (IST), 2010 5th International Symposium on*, pages 812–816.

Tavakoli, H. R., Moin, M. S., and Heikkila, J. (2013). Local similarity number and its application to object tracking. *Int J Adv Robot Syst*, 10(184):1–6.

Yilmaz, A., Javed, O., and Shah, M. (2006). Object tracking: A survey. *ACM Computing Surveys*, 38(14).