Original Article
International Journal of Fuzzy Logic and Intelligent Systems
Vol. 16, No. 2, June 2016, pp. 104-110
http://dx.doi.org/10.5391/IJFIS.2016.16.2.104

ISSN(Print) 1598-2645
ISSN(Online) 2093-744X

Adaptive Bayesian Object Tracking with Histograms of Dense Local Image Descriptors

Minyoung Kim
Department of Electronics & IT Media Engineering, Seoul National University of Science & Technology, Seoul, Korea

Abstract

Dense local image descriptors like SIFT are fruitful for capturing salient information about image, shown to be successful in various image-related tasks when formed in bag-of-words representation (i.e., histograms). In this paper we consider to utilize these dense local descriptors in the object tracking problem. A notable aspect of our tracker is that instead of adopting a point estimate for the target model, we account for uncertainty in data noise and model incompleteness by maintaining a distribution over plausible candidate models within the Bayesian framework. The target model is also updated adaptively by the principled Bayesian posterior inference, which admits a closed form within our Dirichlet prior modeling. With empirical evaluations on some video datasets, the proposed method is shown to yield more accurate tracking than baseline histogram-based trackers with the same types of features, often being superior to the appearance-based (visual) trackers.

Keywords: Computer vision, Object tracking, Bayesian methods, Dense local image descriptors

1. Introduction

Object tracking is the task of localizing regions of the object of interest (e.g., face) in a sequence of image frames, and considered one of the most important problems in computer vision. Recently the dramatic increase of a large amount of video data further demands efficient and accurate tracking algorithms for fast searching, retrieval, and indexing. Although there has been considerable research work conducted for last decades [1, 2], object tracking is still challenging mainly due to the constantly varying appearance of a target object over time, originating from changes in pose, shape, and illumination.

A key component of many state-of-the-art object trackers is the appearance model that represents the very thing that we aim to track. It can be an image template patch itself for the target object [3–6], or alternatively one can use the histogram representation for intensity, color, or edge statistics [7–9]. As the target appearance tends to vary over time (e.g., pose/illumination variation and occlusion), it is crucial to change adaptively the target appearance model, which is typically done by adjusting the model using the recently tracked image patches.

The histogram-based appearance models are beneficial in that they are less sensitive to the

Received: May 31, 2016
Revised : Jun. 17, 2016
Accepted: Jun. 21, 2016

Correspondence to: Minyoung Kim
(mikim@seoultech.ac.kr)

partial occlusion [10]. However, most approaches build histograms using merely the intensity or color statistics, unable to capture higher-order information such as orientations and scales of local gradients (i.e., directions of maximal intensity changes). Motivated by this, in this paper we propose a histogram of the SIFT codewords as a target appearance model. The SIFT [11] descriptors are robust in illumination and viewpoint changes, successful in various tasks including image classification, matching, and annotation [12, 13]. We follow the standard protocol to assign a SIFT codeword to each point in a densely sampled grid. From the computed codewords, it is easy to form a histogram of a candidate tracking region (details shown in Sec. 3 and 4).

Another problem of many existing histogram-based trackers is that they often do not take into account the uncertainty in the chosen target model, but rather take the current model as a ground-truth. The task of target model estimation inherently entails uncertainty due to data noise and model incompleteness, and the tracker would drift eventually unless the uncertainty is properly dealt with. Our second contribution is that we account for uncertainty in the target appearance model by maintaining a distribution over plausible candidate models within the Bayesian framework. Instead of having a single point estimate for the target model, our approach performs a model averaging for prediction, yielding a tracker more robust to noise in observation (e.g., partial occlusion).

Specifically we form a target as a Dirichlet density over the multinomial histogram space where the target compatibility score is defined as an expectation of the histogram similarity score with respect to the current histogram density model. This turns out to be especially beneficial for accounting for uncertainty residing in the target histogram model. Also the target model is updated adaptively by the principled Bayesian posterior inference, which admits a closed form due to our choice of the conjugate prior.

The paper is organized as follows. After briefly formalizing the problem setup with some brief summary of recent related work in Sec. 2, the proposed Bayesian approach of combining base vectors are described in Sec. 3, where we provide experimental evaluations in Sec. 4.

2. Background on Object Tracking

In this section we provide formal problem setup and description for the object tracking problem.

Tracking is an online sequential prediction problem. At each time t, given the image frames $F_0, F_1, \ldots, F_t$ available thus far, and previous tracking decisions $\overline{u}_0, \overline{u}_1, \ldots, \overline{u}_{t-1}$, we make a prediction u_t for the target location in F_t. Here u_t indicates the target state in the image frame F_t, which is, assuming a square axis-parallel target region, represented by three parameters (c_x, c_y, ρ) where (c_x, c_y) is the center position of the target (with respect to the image coordinate in F_t), and $\rho \in (0, \infty)$ is the relative scale compared to the reference size, say (48×48) pixels. Thus u_t determines the cropped image patch for the object, denoted by $I_t = \mathcal{I}(u_t, F_t)$ where $\mathcal{I}(\cdot, \cdot)$ is a well-defined image warping function. Typically the initial state $\overline{u}_0$ in F_0 is given either by a user or an object detection program.

It is common that a tracker maintains the target observation model Θ_t (dependency on t emphasizes that the model can be updated as time goes by), which can be tracker's internal representation for the target appearance, for instance. The observation model naturally defines the goodness (or similarity) measure for a candidate state u_t, namely how much $I_t = \mathcal{I}(u_t, F_t)$ looks like the object that we are going to track. In the simplest first-frame tracker with the fixed target patch $\Theta_t = \overline{I}_0 = \mathcal{I}(\overline{u}_0, F_0)$ regardless of t, the goodness of the state u_t can be defined to be inversely proportional to the exponentiated distance $s(\mathcal{I}(u_t, F_t), I_0) = \exp(-||\mathcal{I}(u_t, F_t) - \overline{I}_0||)$.

Due to the motion smoothness assumption (i.e., the object does not move too far between two consecutive frames), one does not need to search for the best u_t over all possible candidates, but only a small neighborhood centered at the previous track $\overline{u}_{t-1}$, denoted as $\mathcal{N}(\overline{u}_{t-1}) = \{u : ||u - \overline{u}_{t-1}|| \leq \epsilon\}$ for some $\epsilon > 0$. Formally the tracking decision at t can be made by:

$$\overline{u}_t = \arg\max\nolimits_{u_t \in \mathcal{N}(\overline{u}_{t-1})} s(\mathcal{I}(u_t, F_t), \Theta_t) \qquad (1)$$

for a properly chosen tracking similarity measure $s(\cdot, \cdot)$.

Once the tracking decision is made, it results in new data $\overline{I}_t = \mathcal{I}(\overline{u}_t, F_t)$, which can be used to update the observation model, namely $\Theta_{t+1} \leftarrow \text{update}(\Theta_t, \overline{I}_t)$. For instance, in the incremental visual tracker (IVT) [4], the observation model is the low-dimensional PCA subspace built from the previous tracks $\overline{I}_0, \overline{I}_1, \ldots, \overline{I}_{t-1}$, and the subspace update with the new data $\overline{I}_t$ is done by the incremental SVD algorithm [14].

3. Adaptive Bayesian Histogram Tracker

While visual trackers utilize the image patch $\mathcal{I}(u_t, F_t)$ itself (i.e., pixel intensities) to form observation models and distance measures, we rather consider to extract higher-order information using the SIFT descriptors. The SIFT [11] takes into account orientations and scales of the local image gradients in a sophisticated manner, becoming robust to diverse variations in appearance.

For the current image frame F_t, we extract dense SIFT features, say at every 3-by-3 pixel grid point. Each 128-dim SIFT feature vector is then vector quantized into a codeword taking value in $\{1, 2, \dots, K\}$ as follows: For a large collection of SIFT features extracted from an (pre-chosen) image database, we perform K-means clustering to find K cluster centers. That is, the SIFT clustering is done offline. We use the database with thousands of natural scene landscape images collected from the ImageNet image database [15, 16], and the number of clusters is typically set to $K = 100 \sim 150$. Then for each SIFT feature vector from the tracking frame F_t, we find the closest cluster center, and the codeword is determined as its cluster ID.

Now, for the candidate tracking state u_t, we count the numbers of the SIFT codewords that belong to the corresponding patch $I_t = \mathcal{I}(u_t, F_t)$. We let $x_t = [n_1, n_2, \dots, n_K]^\top$ be the frequency count vector where n_j is the number of the codeword j that appears in I_t. Letting $n = \sum_{j=1}^{K} n_j$ (i.e., n is the number of grid points in I_t), the histogram representation for I_t is $h_t = \frac{1}{n} x_t$.

For the target observation model, one may form it as a histogram over the K codewords, namely $\Theta_t = \theta = [\theta_1, \theta_2, \dots, \theta_K]^\top$ with $\sum_j \theta_j = 1$ and $\theta_j \geq 0$ (i.e., θ lies in the $(K-1)$-(probability)-simplex Δ^{K-1}). This modeling choice is tempting in that Θ_t is directly comparable to the candidate h_t, specifically one can adopt as a compatibility score the popular histogram intersection kernel [17, 18]: $s(h_t, \Theta_t) := \sum_j \min((h_t)_j, \theta_j)$. However, this target modeling is a point estimate, and inherently unable to take into account the uncertainty originating from data noise and model incompleteness.

Instead of using the point estimate θ for the target model, we suggest to have a *density* model over all possible histograms. That is, our target model is $P(\theta)$. This accounts for uncertainty in the underlying target histogram, and has an advantageous effect of model averaging. As the density is defined over the $(K-1)$-simplex, we can model it parametrically as the Dirichlet distribution,

$$P(\theta|\alpha) = \text{Dir}(\theta; \alpha) = \frac{1}{B(\alpha)} \prod_{j=1}^{K} \theta_j^{\alpha_j - 1}, \tag{2}$$

where $B(\alpha) = \frac{\prod_{j=1}^{K} \Gamma(\alpha_j)}{\Gamma(\sum_j \alpha_j)}$ with $\Gamma(x) = \int_0^\infty t^{x-1} e^{-t} dt$. The parameter vector $\alpha = [\alpha_1, \dots, \alpha_K]^\top$ is restricted to be positive, and is the very target model we maintain during tracking. In other words, using the notation in Sec. 2, our model at time t is $\Theta_t = \alpha$.

Having the Dirichlet prior is beneficial as it serves as a conjugate prior in conjunction with the multinomial likelihood. That is, at time t when we have the decision of the target containing the codeword counts, $x_t = [n_1, n_2, \dots, n_K]^\top$ with $\sum_j n_j = n$, the posterior distribution of the model follows the Bayes rule $P(\theta|x_t) \propto p(x_t|\theta)P(\theta)$, and using the multinomial $P(x_t|\theta) = \frac{n!}{n_1! \cdots n_K!} \prod_j \theta_j^{n_j}$, it also becomes Dirichlet:

$$P(\theta|x_t) = \text{Dir}(\theta; \alpha') \ \text{ where } \ \alpha'_j = n_j + \alpha_j, \ \forall j. \tag{3}$$

The equation (3) becomes our model update equation, namely $\Theta_{t+1} = \alpha' = x_t + \alpha$ (addition elementwisely).

Next we need to define the compatibility score for a candidate track $x_t = [n_1, n_2, \dots, n_K]^\top$ with $\sum_j n_j = n$ at time t with respect to the current histogram density model $P(\theta; \alpha)$. Simply considering the marginal likelihood $p(x_t|\alpha)$ as a compatibility score turns out to be problematic. This is because:

$$\begin{aligned} P(x_t|\alpha) &= \int P(x_t|\theta)P(\theta;\alpha)d\theta & (4) \\ &= \frac{n!}{n_1! \cdots n_K!} \frac{1}{B(\alpha)} \int \prod_{j=1}^{K} \theta_j^{n_j + \alpha_j - 1} d\theta & (5) \\ &= \frac{B(x_t + \alpha)}{B(\alpha)} \frac{n!}{n_1! \cdots n_K!}, & (6) \end{aligned}$$

and $P(x_t|\alpha)$ may have highly different scales across different candidates x_t's as the numbers of SIFT grid points in target patches (n) can vary (e.g., consider target patches of highly different sizes).

Instead we propose a compatibility score based on the *expected histogram intersection kernel*. The idea is to average the histogram similarities between the target and *all* plausible histograms from the current model. For (unnormalized) x_t we form a histogram $h_t = \frac{1}{n} x_t$, and define the similarity between the candidate x_t and the current model α by the expected histogram kernel, namely

$$\begin{aligned} s(h_t, \alpha) &= \mathbb{E}_{P(\theta|\alpha)}[s(h_t, \theta)] & (7) \\ &= \int P(\theta|\alpha) s(h_t, \theta) d\theta & (8) \\ &\approx \frac{1}{N} \sum_{i=1}^{N} s(h_t, \theta^{(i)}), & (9) \end{aligned}$$

where in (9) we do Monte Carlo (MC) approximation for the difficult integration by sampling N iid samples $\{\theta^{(i)}\}_{i=1}^{N}$ from the current model $P(\theta|\alpha)$.

In summary, our target appearance model is the Dirichlet distribution over the histograms $P(\theta|\alpha)$, and the similarity score $s(h_t, \alpha)$ for a candidate target histogram h_t is the expected (or MC-averaged) histogram intersection kernel over the current Dirichlet histogram density. Once the best candidate x_t is found, we update our model by incorporating the sample just found. By the Bayes rule, we have the posterior in the same Dirichlet family, namely $P(\theta|x_t) = \text{Dir}(\theta; x_t + \alpha)$.

4. Experiments

Now we test the proposed adaptive Bayesian histogram tracker on three real-world videos from `http://cvlab.hanyang.ac.kr/tracker_benchmark/` (Box, Motor rolling, and Bolt where some sample frames are depicted in Figure 1). As mentioned earlier, for many natural scene images (at offline) we extract and build a pool of SIFT features, from which we learn $K = 150$ cluster center vectors from the K-means clustering. In this way, for a candidate image patch during tracking, we can form a histogram representation (the number of bins is $K = 150$) for the patch by performing vector quantization (i.e., imputed to the closest cluster center ID).

At each frame during tracking, we set the search space as a bounding box centered at the previous target location and twice the size of the previous target region. Within the bounding box, we randomly generate 300 axis-parallel rectangular candidate tracks, among which we compute the compatibility scores with respect to the tracker's target model. The candidate with the highest score is then chosen, and subsequently used to update the tracker.

For comparison, we consider the baseline histogram tracker that maintains the point estimate histogram θ as the target model, and update the model by the simple exponential smoothing (averaging) of the all historic samples (with emphasis more on recent ones). That is, for the histogram h_t of the best candidate in the current frame, we update: $\theta^{new} = (1 - \eta)\theta + \eta h_t$ where $\eta \in [0, 1]$ is parameter of forgetting factor. We choose empirically $\eta = 0.95$. We also compare our tracker with the popular incremental visual tracking algorithm (IVT) [4] that maintains the PCA subspace of the tracked target image patches, thus not histogram-based.

For quantitative performance comparison, we record the tracking drifts measured as Euclidean distance (in pixels) between the centers of the ground-truth target and the tracked one. In Table 1 we summarize the per-frame averaged drifts for three competing methods: simple exponential smoothed histogram tracker (Smooth-Hist), the PCA-based incremental visual tracker (IVT) [4], and our proposed Bayesian adaptive histogram tracker (Bayes-Hist). As shown, the Bayes-Hist consistently attains superb tracking performance than the visual tracking and the baseline histogram tracker. This signifies the impact of incorporating salient SIFT features in bag-of-words forms into a tracker's target model as well as the Bayesian model averaging that effectively accounts for uncertainty originating from data noise and model incompleteness.

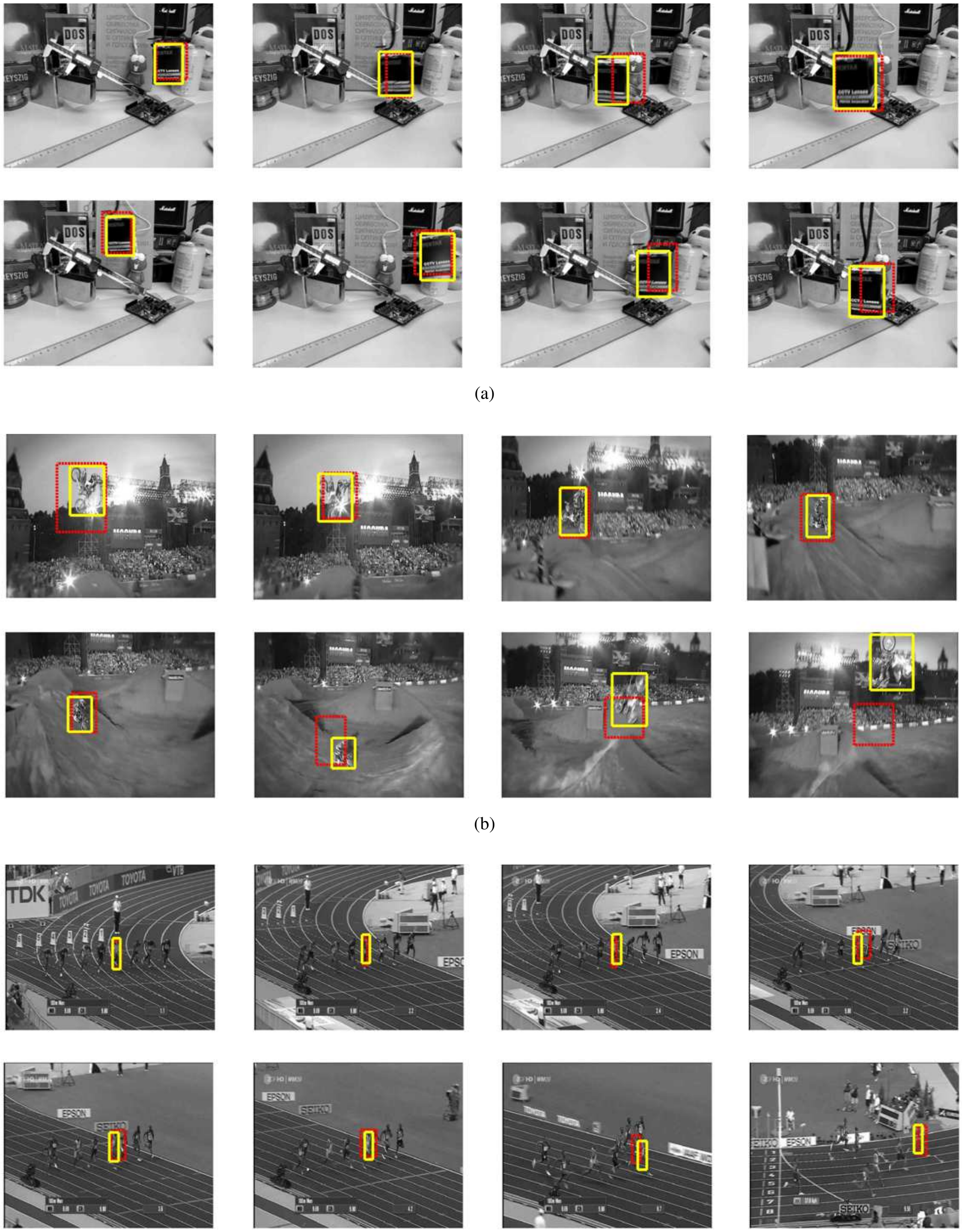
Table 1. Per-frame average tracking errors (in pixels)

	Box	Motor rolling	Bolt
IVT	12.59	15.26	7.56
Smooth-Hist	16.69	29.33	5.03
Bayes-Hist	4.48	5.16	3.78

It is interesting to note that in the Bolt video, we see that the histogram-based trackers, event the baseline Smooth-Hist, yield smaller tracking errors than the visual tracker. This can be explained as follows: the target appearance contains highly distinct texture (i.e., specific repeating intensity patterns) against the background, which is well captured and discriminated by the dense SIFT features than just intensity values alone. For some selected frames we also depict the tracking results contrasting the proposed method and the IVT in Figure 1.

5. Conclusion

In this paper we have proposed a novel Bayesian adaptive histogram tracker that incorporates the dense local image descriptors in an effective way. The Bayesian model averaging performed during tracking decision turns out to yield more robust and accurate tracking results than existing visual tracking methods. One potential caveat of the proposed approach is that extracting SIFT features for every frame is computationally intensive, and achieving real-time tracking in practice needs more technical remedy. For instance, one can address the issue to some extent by either restricting the search space more aggressively, or delaying the tracking decision for several forthcoming frames and pipelining SIFT extraction for newly coming frames. Other ways of possible computational speedup will be pursued in our future work.



(a)

(b)

(c)

Figure 1. Tracking results on three videos. (a) Box, (b) Motor rolling, and (c) Bolt. Selected frames are highlighted where the yellow-solid box indicates our Bayesian histogram tracker, while the red-dashed is IVT. The ground-truths are not shown, but they are tight bounding boxes around the objects, which can be easily inferred visually.

Conflict of Interest

No potential conflict of interest relevant to this article was reported.

Acknowledgments

This study was supported by Seoul National University of Science & Technology.

References

[1] K. Zhang, Q. Liu, Y. Wu, and M. H. Yang, "Robust visual tracking via convolutional networks without training," *IEEE Transactions on Image Processing*, vol. 25, no. 4, pp. 1779-1792, 2016. http://dx.doi.org/10.1109/TIP.2016.2531283

[2] J. H. Yoon, M. H. Yang, and K. J. Yoon, "Interacting multiview tracker," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 5, pp. 903-917, 2016. http://dx.doi.org/10.1109/TPAMI.2015.2473862

[3] M. J. Black and A. D. Jepson, "EigenTracking: Robust matching and tracking of articulated objects using a view-based representation," *International Journal of Computer Vision*, vol. 26, no. 1, pp. 63-84, 1998. http://dx.doi.org/10.1023/A:1007939232436

[4] D. A. Ross, J. Lim, R. S. Lin, and M. H. Yang, "Incremental learning for robust visual tracking," *International Journal of Computer Vision*, vol. 77, no. 1, pp. 125-141, 2008. http://dx.doi.org/10.1007/s11263-007-0075-7

[5] M. Kim, "Correlation-based incremental visual tracking," *Pattern Recognition*, vol. 45, no. 3, pp. 1050-1060, 2012. http://dx.doi.org/10.1016/j.patcog.2011.08.026

[6] L. Sevilla-Lara and E. Learned-Miller, "Distribution fields for tracking," in *Proceedings of 2012 IEEE Conference on Computer Vision and Pattern Recognition*, Providence, RI, 2012, pp. 1910-1917. http://dx.doi.org/10.1109/CVPR.2012.6247891

[7] D. Comaniciu, V. Ramesh, and P. Meer, "Kernel-based object tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 5, pp. 564-577, 2003. http://dx.doi.org/10.1109/TPAMI.2003.1195991

[8] A. Adam, E. Rivlin, and I. Shimshoni, "Robust fragments-based tracking using the integral histogram," in *Proceedings of 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, New York, NY, 2006, pp. 798-805. http://dx.doi.org/10.1109/CVPR.2006.256

[9] S. He, Q. Yang, R. W. H. Lau, J. Wang, and M. H. Yang, "Visual tracking via locality sensitive histograms," in *Proceedings of 2013 IEEE Conference on Computer Vision and Pattern Recognition,* Portland, OR, 2013, pp. 2427-2434. http://dx.doi.org/10.1109/CVPR.2013.314

[10] A. Bolovinou, I. Pratikakis, and S. Perantonis, "Bag of spatio-visual words for context inference in scene classification," *Pattern Recognition*, vol. 46, no. 3, pp. 1039-1053, 2013. http://dx.doi.org/10.1016/j.patcog.2012.07.024

[11] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91-110, 2004. http://dx.doi.org/10.1023/B:VISI.0000029664.99615.94

[12] L. Fei-Fei and P. Perona, "A Bayesian hierarchical model for learning natural scene categories," in *Proceedings of 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, San Diego, CA, 2005, pp. 524-531. http://dx.doi.org/10.1109/CVPR.2005.16

[13] W. Chong, D. Blei, and F. F. Li, "Simultaneous image classification and annotation," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, Miami, FL, 2009, pp. 1903-1910. http://dx.doi.org/10.1109/CVPR.2009.5206800

[14] A. Levey and M. Lindenbaum, "Sequential Karhunen-Loeve basis extraction and its application to images," *IEEE Transactions on Image Processing*, vol. 9, no. 8, pp. 1371-1374, 2000. http://dx.doi.org/10.1109/83.855432

[15] J. Deng, W. Dong, R. Socher, L. J. Li, K. Li, and L. Fei-Fei, "ImageNet: a large-scale hierarchical image database," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition,* Miami, FL, 2009, pp. 248-255. http://dx.doi.org/10.1109/CVPR.2009.5206848

[16] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, et al, "ImageNet large scale visual recognition challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211-252, 2015. http://dx.doi.org/10.1007/s11263-015-0816-y

[17] A. Barla, F. Odone, and A. Verri, "Histogram intersection kernel for image classification," in *Proceedings of International Conference on Image Processing*, Barcelona, Spain, 2003, pp. 513-516. http://dx.doi.org/10.1109/ICIP.2003.1247294

[18] O. Pele and M. Werman, "The quadratic-chi histogram distance family," in *Proceedings of 11th European Conference on Computer Vision*, Crete, Greece, 2010, pp. 749-762. http://dx.doi.org/10.1007/978-3-642-15552-9_54

Minyoung Kim received his B.S. and M.S. degrees both in computer science and engineering from Seoul National University, South Korea. He earned a Ph.D. degree in computer science from Rutgers University in 2008. From 2009 to 2010 he was a postdoctoral researcher at the Robotics Institute of Carnegie Mellon University. He is currently an assistant professor in the Department of Electronics and IT Media Engineering at Seoul National University of Science and Technology in Korea. His primary research interest is machine learning and computer vision. His research focus includes graphical models, motion estimation/tracking, discriminative models/learning, kernel methods, and dimensionality reduction.