https://doi.org/10.2352/ISSN.2470-1173.2020.9.IQSP-169

Quality Aware Feature Selection for Video Object Tracking

Roger Gomez Nieto, Hernan Dario Benitez Restrepo
Department of Electronics and Computer Science, Pontificia Universidad Javeriana, Cali, Colombia.
José Francisco Ruiz-Muñoz
Department of Electrical and Computer Engineering, University of Florida, Gainesville, FL, USA.

Abstract

Video object tracking (VOT) aims to determine the location of a target over a sequence of frames. The existing body of work has studied various image factors that affect VOT performance. For instance, factors such as occlusion, clutter, object shape, unstable speed and zooming, that influence video quality, do affect tracking performance. Nonetheless, there is no clear distinction between scene-dependent challenges such as occlusion and clutter and the challenges imposed by traditional notions of "quality impairments" inherited from capture, compression, processing, and transmission. In this study, we are concerned with the latter interpretation of quality as it affects video tracking performance. In this paper, we propose the design and implementation of a quality aware feature selection for VOT. First, we divided each frame of the video into patches of the same size and extracted HOG, and natural scene statistics (NSS) features from these patches. Then, we degraded the videos synthetically with different levels of post-capture distortions such as MPEG-4, AWGN, salt and pepper, and blur. Finally, we defined the best set of features HOG and NSS that generate the largest area under the curve in the success plots, yielding an improvement in the video tracker performance in videos affected by post-capture distortions.

Introduction

Video Object Tracking (VOT) is a complex process that allows to locate and follow one object over time using video streaming. Recently, many works present different approaches to solve this challenge. However, to the best of our knowledge, any of these approaches has not modeled and quantified the influence of post-capture distortions on the object tracking performance. This question is very important because state-of-the-art trackers perform well in videos with few or non-distortions, but when they are tested on videos affected by distortions, such as those acquired by surveillance cameras, the performance can be degraded to a large extent.

We proposed an approach to integrate NSS perceptual quality features into a video object tracker scheme and demonstrated its performance in several videos affected by post-capture distortions. Previous works on this topic have focused on tasks such as object and face detection [1], dermoscopy [2], and face recognition in long-wave infrared (LWIR) images [3][4].

Dai et al. [5] studied the influence of the presence of shaking motions in VOT. They found that trackers fail by this distortion because of two main issues. The first one occurs when the whole tracked target is not found in the candidates patches because it has moved out fast due to significant shaking motion. The second one is that the shaking movement may generate blur distortion, and trackers are confused by blurred boundaries between foreground and background.

To the best of our knowledge, this is the first work to propose a quality-aware feature extraction approach in VOT. Furthermore, this approach complements our previous work in which we demonstrated the impact of authentic distortions on state-of-the-art video trackers [6]. This paper is structured as follows. First, we present the VOT scheme and the HOG and NSS features. Second, we explain the the results and discussion and finally we present the conclusions.

Materials and Methods

Video Object Tracker

We present a video object tracker framework based on a support vector machine (SVM)[7–9]. Figure 1 shows the diagram of our approach. Basically, this tracker consists of the classification between target and background of patches represented in a feature space [10].

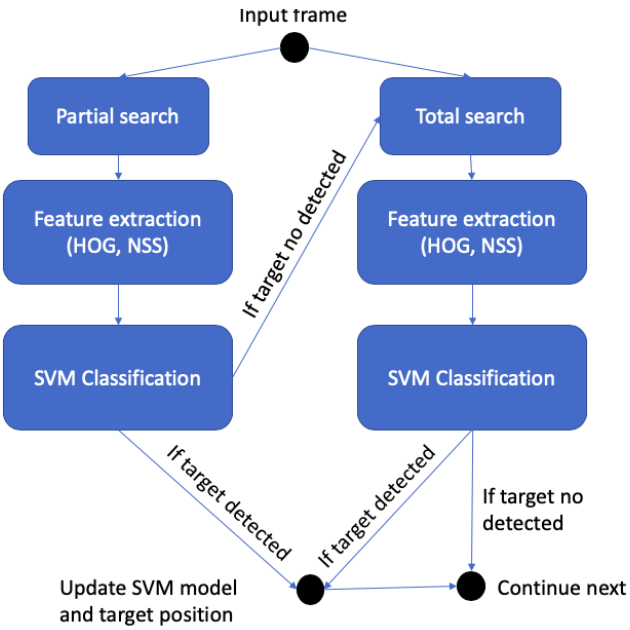


Figure 1: State diagram of the proposed video object-tracking framework.

For starting the tracking, our algorithm requires the bounding box that indicates the location of the target in the first frame. The initial training set is generated as follows: as target class examples, we set the number of target objects $N_{tar} = 10$. These target objects are computed from the first bounding box and its

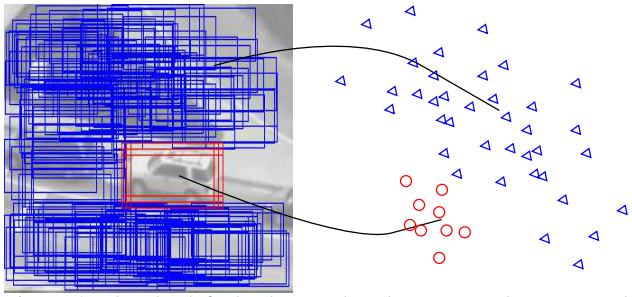


Figure 2: On the left, background and target patches `bgP` and `objP` in blue and red, respectively.

neighborhood (random patches that overlap with at least 80% the target bounding box). Likewise, as background examples, we randomly select N_{bg} patches (not overlapped with the target). Figure 2 illustrates the target and background patches used to generate the training set.

In our experiments, N_{bg} corresponds to 10% of the feature representation. We use this data to train a linear two-class SVM classifier. On the next frame, we carry out a "partial search" of the target. For this purpose, we find a set of query patches from the image region that contains the bounding boxes that overlap with at least 50% with the bounding box of the target in the previous frame. Afterward, we classify the query objects with the current SVM classifier. If one patch is classified as target, we update the location accordingly. If more than one patch is classified as target, we interpolate the bounding boxes of all them. If the target is not found, we carry out a "total search." The total search consist of the classification of all the patches (with the same size of the target) in the whole image. If at least one patch is classified as target, the process to update the location is the same as in the partial search. After updating the target location, we also update the training set by choosing new target and background examples as in the first frame. The maximum size of the training set is the feature dimension. When the data set exceeds the maximum size, we get rid of the oldest examples. If the target is not found in the total search either, we move to the next frame without updating the location nor the training set.

To evaluate the video tracker performance, we used success plots, inspired by the work presented in [11]. To generate a success plot, we calculate an overlap score S for each frame of a video, defined as

$$S = \frac{|r_t \cap r_o|}{|r_t \cup r_0|}, \tag{1}$$

where r_t denotes the object bounding box estimated by the tracker, r_0 is the ground-truth bounding box, $\cap$ and $\cup$ are the intersection and union operators respectively, and $|\cdot|$ represents the number of pixels within a region. Figure 3 shows a graphic description of the regions defining S.

We define the area under the curve (AUC) as the trapezoidal integral of the success plot (i.e S along the whole video). AUC lies in the range [0,1] and values closer to 1 are better.

Feature extraction

For the analysis in the vector space (SVM classifier training and test), we represent a patch in the video frame as feature vector

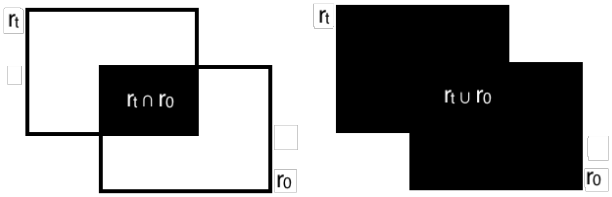


Figure 3: Definition of the regions used for calculating the overlap score S.

$x \in \mathbb{R}^d$ (where d is the dimension of the space), by two type of descriptors:

- **Histogram of Oriented Gradients (HOG)**: a 1-D histogram is computed from the gradient directions in a small region of an image [12]. This region is called a "cell." To reduce the effect of the illumination changes over the image, each cell is normalized by the total energy in a set of neighbor cells called "block." We take as the HOG feature representation $x_{HOG} \in \mathbb{R}^{d_{HOG}}$, the average of the histograms of all the cells inside the corresponding bounding box. The dimension of this representation is given by the number of histogram bins d_{HOG}.
- **Natural Scene Statistics in the spatial domain (NSS)** [13]: We compute the NSS feature representation $x_{NSS} \in \mathbb{R}^{d_{NSS}}$ of a patch by the 36 features extracted from locally normalized luminances as described in [13].

For avoiding an undesired effect induced by the scale of the features, we consider two normalization methods for each type of descriptors:

- z-score: the mean μ and standard deviation σ of each feature distribution are made 0 and 1, respectively as follows

$$\bar{x}_z = \frac{x - \mu_o}{\sigma_o},$$

  where the original mean μ_o and standard deviation σ_o are estimated from the training set.
- 0-1-normalization: each feature distribution is re-scaled in order to set 0 as the minimum value and 1 as the maximum value:

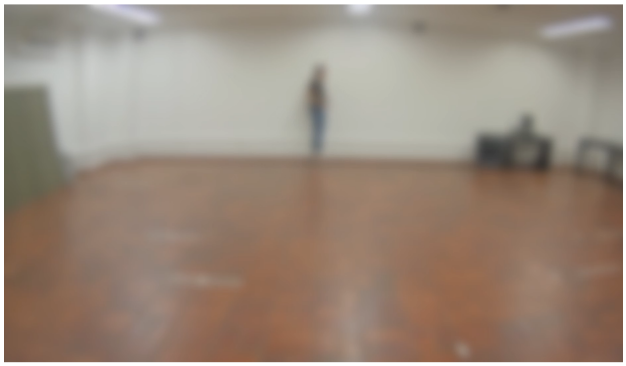
$$\bar{x}_n = \frac{x - x_{\min}}{x_{\max}}.$$

Results and Discussion

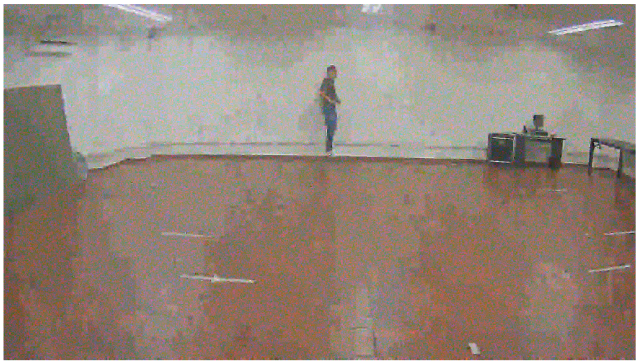
In this section, we show the object tracking results obtained with HOG and HOG+NSS method, tested on 910 videos of the constructed dataset. Our Quality-Aware tracker is implemented with Matlab 2018a and all the experiments are run on a PC equipped with Intel i7 8750-H CPU, 32GB RAM and a single NVIDIA GTX 1060 GPU.

Image distortions

We consider four basic types of distortions that commonly occur in digital devices and over communication channels. The distortions here used are related to the encoding (compression) and transmission processes (post-capture distortions) [1].

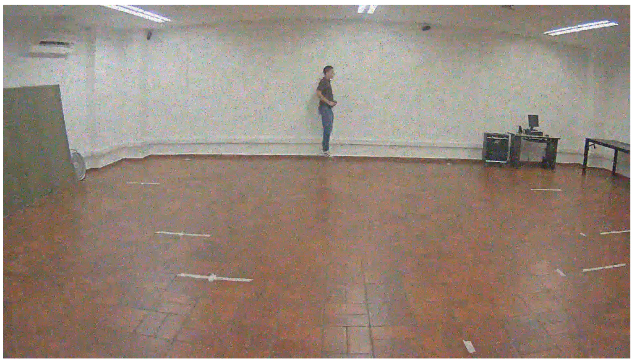


(a) Blur



(b) AWGN

(c) MPEG-4



(d) SAP

Figure 4: Distorted frames with high-level intensities.

Figure 5: Pristine frame of proposed dataset, person jumping in indoor environment.

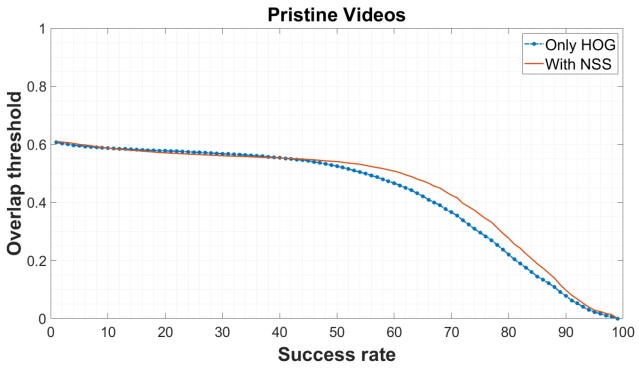
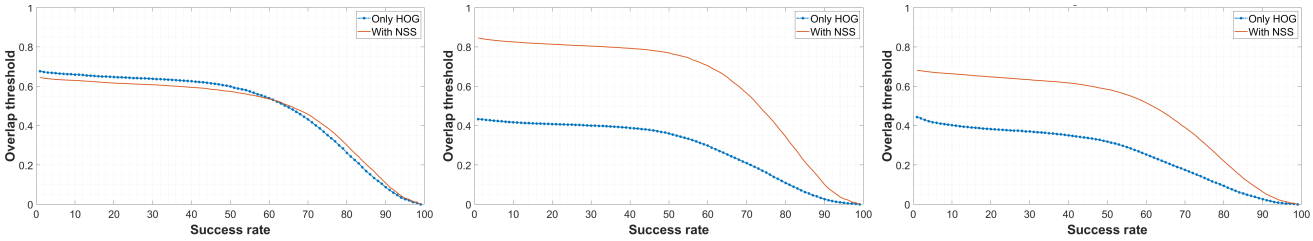
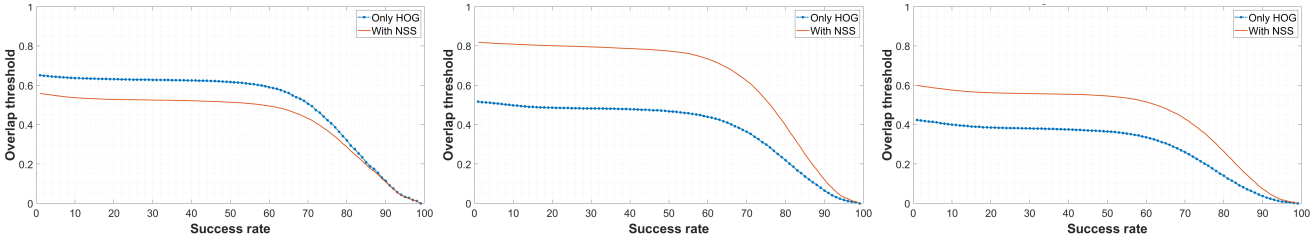


Figure 6: Results in pristine videos of the proposed dataset. The red line is the performance of quality-aware method using NSS features. The blue dotted line is the performance of only HOG features representation.

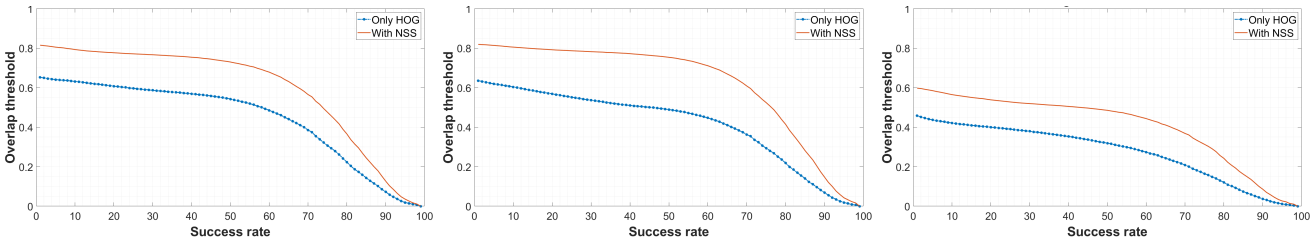
- **AWGN**, Additive White Gaussian Noise: This is a local distortion, in which a zero-mean Gaussian noise of variance parameter is added independently to each pixel. The imnoise() function in MATLAB was used to introduce additive white Gaussian noise to the images. Three levels of AWGN were added with the noise variance parameters equal to [0.01, 0.05, 0.1].
- **Blur**: This is a global distortion in which each pixel is blurred through convolution with a gaussian low pass filter of standard deviation. The imfilter() function in MATLAB was used to introduce Gaussian blur at three levels. The standard deviation of the Gaussian filter was varied over a log scale, σ_B = [5, 10, 15].
- **Salt and Pepper noise (SAP)**: This distortion generates only a few very noisy pixels. The effect is similar to sprin-



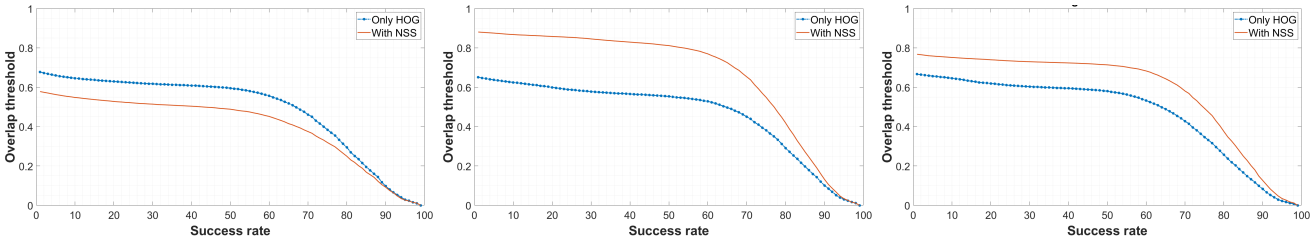
(a) Low intensity Blur (b) Medium intensity Blur (c) High intensity Blur



(d) Low intensity AWGN (e) Medium intensity AWGN (f) High intensity AWGN



(g) Low intensity MPEG-4 (h) Medium intensity MPEG-4 (i) High intensity MPEG-4



(j) Low intensity SAP (k) Medium intensity SAP (l) High intensity SAP

Figure 7: The overlap success plots of Quality-Aware tracker tested on videos with several post-capture distortions and intensity levels. The red line is the performance of quality-aware method using NSS features. The blue dotted line is the performance of only HOG features representation.

kling white and black dots—salt and pepper—on the image. One example where salt and pepper noise arises is in transmitting images over noisy digital links [14]. The imnoise() function in MATLAB was used to introduce SAP noise to the images. We added three levels of SAP noise, being D the noise density. This noise affects approximately $D * numel(I)$ pixels. The used values for D are [0.01, 0.05, 0.1], for high, medium, and low levels, respectively.

- **MPEG-4** compression: The MPEG-4 standard is used in a wide variety of video applications, such as DVDs and digital broadcast television. Compression systems such as MPEG-4 produce relatively uniform distortions/quality in the video, both spatially and temporally. MPEG-4 compressed videos exhibit typical compression artifacts such as blocking, ringing, and motion compensation mismatches around object edges [15]. To generate this distortion, we used ffmpeg[1] tool with a bitrate of 100Kbps, 200Kbps, and 1Mbps, for high, medium, and low levels, respectively.

Proposed dataset

We recorded 70 pristine videos with activities such as walking, jumping, and sitting. We chose these activities because they are simple, with only one target in the scene. This condition allows us to test fairly several trackers in such a way that the most critical challenge was the distortions applied. The videos were recorded using four surveillance cameras in indoor environments. These pristine videos were impaired with four post-capture distortions: blur, AWGN, MPEG-4 and SAP. All the these distortions have three intensity levels. Hence, the whole dataset have 910 videos (70 pristine, and 210 per each distortion). We have made available this dataset to the scientific community at `https://bit.ly/2vahVgC`. Figure 6 shows one pristine video frame and Figure 4 present the same frame affected by the four distortions at the highest level.

Figure 7 represents the average success plots from 70 videos corresponding to one level of distortion (low, medium, and high). HOG+NSS representation performs significantly better than only HOG features, except when the distortion is low. This difference is due to the specialization of our method in highly distorted videos indicating that the introduction of NSS features is more helpful in VOT task with highly distorted videos. To demonstrate this idea, Figure 6 shows the performance on pristine videos. This performance is almost equivalent between both methods, being slightly better the HOG+NSS method.

Statistical significance test

Since non-parametric tests make no assumptions about the probability distributions of the variables, we conducted a Kruskal-Wallis test on each median values of AUC for 70 videos comparing the HOG and HOG+NSS methods to evaluate whether the results presented in 7 are statistically significant. Table 1 tabulates the results of the statistical significance test. From Table 1, we conclude that HOG+NSS method produced highly competitive object tracking performance on the tested videos with statistical significance against the only HOG algorithm tested.

[1] FFmpeg Developers, (2016). Available from http://ffmpeg.org/

Table 1: Statistical significance matrix of AUC values between HOG and HOG+NSS . A value of "1" indicates that the performance of the model with NSS was statistically better than that of the model with only HOG, "0" means that it is statiscally worse, and "-" means that it is statistically indistinguishable

	pristine	MPEG-4	S&P	Blur	Gaussian
Statistical significance	-	1	1	1	1

Conclusions and Future Work

The results obtained by selecting and incorporating quality aware features into the representation of the image patches show an improvement in the VOT performance, in terms of AUC, for blur, SAP, AWGN, and MPEG-4 post-capture and encoding distortions. The performance loss in unconstrained VOT can be due to several scene conditions such as occlusion, scale change, and cluttering which are different from image quality degradations. In this paper, we only focused on VOT for scenes without occlusions and scale changes to isolate the loss in performance due to image quality, while we primarily considered only post-capture distortions such as blur, AWGN, MPEG4, and SAP here, it would be of interest to study other distortions due to over or under exposure and images authentically distorted by a combination of multiple artifacts when captured with a camera.

Acknowledgments

The authors acknowledge the funding provided by COLCIENCIAS Colombia and Pontificia Universidad Javeriana with the project *Vigilancia Inteligente para la red de cámaras de la Policía Metropolitana de Cali*. The authors would like to thank NVIDIA Corporation for the donation of a TITAN XP GPU used in these experiments.

References

[1] S. Gunasekar, J. Ghosh, and A. C. Bovik. Face detection on distorted images augmented by perceptual quality-aware features. *IEEE Trans. Inf. Forensics Security*, 9:2119–2131, 2014.

[2] F. Xie, Y. Lu, A.C Bovik, Z. Jiang, and R. Meng. Application-driven no-reference quality assessment for dermoscopy images with multiple distortions. *IEEE Transactions on Biomedical Engineering*, 63(6):1248–1256, 2016.

[3] R. Soundararajan and S. Biswas. Machine vision quality assessment for robust face detection. *Signal Process. Image Communication*, 72:92–104, 2019.

[4] C. G. Rodriguez-Pulecio, H. D. Benitez-Restrepo, and A. C. Bovik. Making long-wave infrared face recognition robust against image quality degradations. *Quant. Infrared Thermogr. Journal*, 16(3-4):218–242, 2019.

[5] Manna Dai, Shuying Cheng, Xiangjian He, and Dadong Wang. Object tracking in the presence of shaking motions. *Neural Computing and Applications*, 31(10):5917–5934, 2019.

[6] R. Gomez-Nieto, H. D. Benitez-Restrepo, and I. Cabezas. How video object tracking is affected by in-capture distortions? In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2019)*, 2019.

[7] Matthew B Blaschko and Christoph H Lampert. Learning to localize objects with structured output regression. In *European Conference on Computer Vision*, pages 2–15. Springer,

2008.

[8] Andrea Vedaldi, Varun Gulshan, Manik Varma, and Andrew Zisserman. Multiple kernels for object detection. In *2009 IEEE 12th International Conference on Computer Vision*, pages 606–613. IEEE, 2009.

[9] Pedro F Felzenszwalb, Ross B Girshick, David McAllester, and Deva Ramanan. Object detection with discriminatively trained part-based models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(9):1627–1645, 2010.

[10] Shai Avidan. Support vector tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(8):1064–1072, 2004.

[11] Y. Wu, J. Lim, and M. H. Yang. Object tracking benchmark. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(9):1834–1848, Sept 2015.

[12] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*, volume 1, pages 886–893. IEEE, 2005.

[13] Anish Mittal, Anush Krishna Moorthy, and Alan Conrad Bovik. No-reference image quality assessment in the spatial domain. *IEEE Transactions on image processing*, 21(12):4695–4708, 2012.

[14] Alan C Bovik. *The essential guide to image processing*. Academic Press, 2009.

[15] Kalpana Seshadrinathan, Rajiv Soundararajan, Alan Conrad Bovik, and Lawrence K. Cormack. Study of subjective and objective quality assessment of video. *IEEE Transactions on Image Processing*, 19(6):1427–1441, jun 2010.

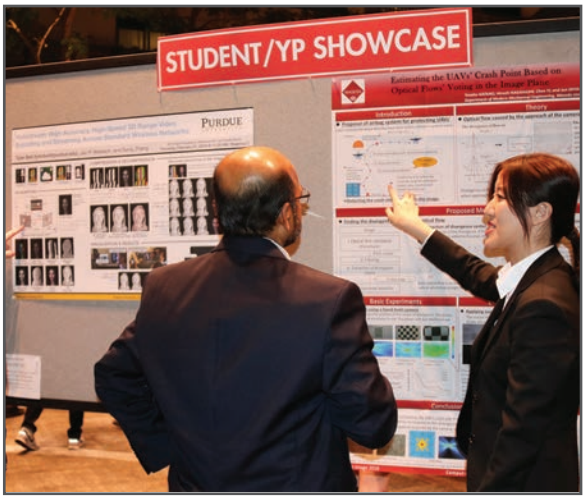
JOIN US AT THE NEXT EI!

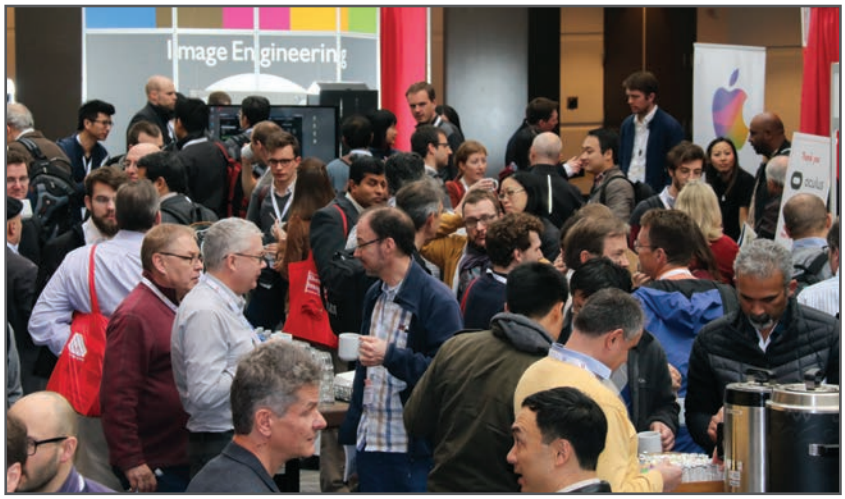
IS&T International Symposium on

Electronic Imaging

SCIENCE AND TECHNOLOGY

Imaging across applications . . . Where industry and academia meet!





• SHORT COURSES • EXHIBITS • DEMONSTRATION SESSION • PLENARY TALKS •
• INTERACTIVE PAPER SESSION • SPECIAL EVENTS • TECHNICAL SESSIONS •

www.electronicimaging.org

IS&T
imaging.org