Large-scale forensic analysis of security images and videos

Craig Henderson[1]
c.d.m.henderson@qmul.ac.uk
Prof. Ebroul Izquierdo[1]
http://www.eecs.qmul.ac.uk/~ebroul/

[1] Multimedia and Vision Lab
Queen Mary University of London
London, UK

Abstract

Our research is concerned with the practical application of computer vision in the forensic analysis of security images and videos. Contemporary literature make use of high-definition images and Hollywood feature films in their datasets, and there is little or no assessment of algorithms' performance using poor quality images with variable frame rates and uncontrolled lighting conditions such as security video.

Work so far has produced a methodology for matching features across low quality images that yields an improved results over existing feature matching techniques. Future work will involve innovation in search and retrieval online machine learning to train models from unlabelled data, and segmentation of one-shot videos to aid computer and human analysis of long-running video sequences. We are motivated to produce an integrated system for police investigators to use a query-by-example search and retrieval system with relevance feedback and machine learning to incrementally discover evidence in criminal investigations.

1 Introduction

Research in the computer analysis of images and video sequences is prevalent in contemporary literature, and advances are frequent and substantial. However, the bulk of the research studies high quality imaging, or devises algorithms based upon knowledge of the scene, such as camera position or imaging conditions. Working with the Metropolitan Police in London has demonstrated that the opportunity to use existing computer vision techniques in the analysis of real-world security video such as closed-circuit television (CCTV) is severely limited because of the poor quality of the video, and the lack of metadata. These limitations are sometimes so fundamental that they would be inconceivable without understanding the current practice of video acquisition during the course of a police investigation.

The source of video footage can be varied, as can the quality. There is a lack of standardisation in security camera systems. Obtained footage is often in a proprietary format that can only be viewed onscreen by a manufacturer-supplied application, and the quality and usability of these applications vary tremendously. To achieve their goal of carefully watching and re-watching segments of video, and to be able to edit videos into a

story that can be used in court, the Met Police have employed creative solutions to overcome the limitations of the source video images. The result is a time-, machine- and human-intensive activity to transcode the video footage by *recapturing* the video played on a computer screen. The video is played onscreen in real-time and the screen presentation is recorded to a standard video format that can be viewed and edited as required, and can also then be used in computer vision applications and research.

The frame rate of a video is measured by the number of frames per second, fps, that are captured. A lower frame rate means there is a greater difference (movement) in adjacent frames and features are further away relative to the previous frame and move greater distances relative to each other. Adjacent frames therefore have a greater visual difference than those from a high frame rate video. This difference can significantly affect the robustness of computer vision algorithms that often rely on assumptions that two consecutive frames in an image are very similar, and that the movement between features in the frame can be used as prior knowledge in the design of an algorithm. For example, a feature tracking algorithm makes the determination of whether features are related or not based on the amount of global and relative movement between frames, known as *spatial consistency*, and assumes that the movement is consistent through the video. In a low frame rate video, such determination is less robust as the movement threshold must be increased to compensate for the additional movement, and this increase can introduce noise and misclassifications. It would be feasible to configure spatial consistency algorithms using a video's metadata, for example to adapt the spatial distance threshold of related features based on the frame rate of the video. In our area of interest, security videos very often have no associated metadata, and therefore cannot be used as a reliable input into algorithmic choices for spatial consistency parameters.

Recapturing enables the police to use standard tools to watch and analyse the video, and provides an opportunity for automated analysis. However, useful metadata that one would expect to be available in a video file, such as the video frame rate and date/time stamps, is missing or, perhaps worse, inaccurate. For example, recapturing records at a fixed frame rate, perhaps 25 fps, and this is recorded in the metadata of the resulting video. If the video being played is at a lower frame rate, then multiple frames will be captured for each frame in the original video file. The playback is visually unaffected, but duplicate consecutive frames add another complication for computer vision applications as the amount of movement between pairs of adjacent frames is inconsistent. Another metadata is time sequence. Time sequences would enable software to synchronise video captured from multiple cameras based upon the time associated with the video sequence. Edelman [1] reported a system at the Netherlands Forensic Institute which uses Optical Character Recognition to read video timestamps from the video images. Such a technique is not reliable enough to provide sufficient metadata for steering Computer Vision algorithms, however. In addition, the Met Police observe that camera timestamps are unreliable as the accuracy of the time is dependent on the ongoing maintenance of the security system, and varies considerably between local authority, police and private owners.

Forensic analysis of security camera video is a less well studied field and demands adaptation of contemporary methods to accommodate the image quality variances that exist. Images are often low resolution with poor colour clarity and have little discriminative or representative texture definition. Security video is often passively recorded by a fixed or pre-defined motion path camera with the camera mounted on a

bracket that automatically moves around a loop or figure-of-eight to provide maximum coverage of an area with a single camera. Objects will therefore move in and out of view regularly within a sequence of frames. Other cameras are human-operated and can record very erratic movement with dramatic changes of focus and rapid zoom as the camera operator wrestles with the controls to record the action on the streets. The fast movement in pan and zoom, either in the manually controlled camera or to a lesser extent in a fixed-path motion camera degrades the image quality further with motion blur, and is somewhat unique to the security videos such as those that we analyse. Quality is further reduced by varying weather conditions where the changes in light, presence of rain, snow, mist or fog, direct sunlight and shadows can all affect the clarity of an image, and the ability for a feature extractor to consistently describe an image region.

Query-by-example search and retrieval

We aim to develop methods of identifying and matching distinctive regions or patterns across frames of security camera video sequences. Established methods of feature detection, extraction and matching perform less well with low quality images than in high-definition images with sharp focus and controlled lighting conditions, such as Hollywood films popularly used in the literature [2], [3], [4].

Given a user-specified rectangular region of interest, our system will identify distinctive regions or patterns and trace them through disparate surveillance videos, using a combination of feature tracking and search-and-retrieval techniques. The goal is to identify video sequences that contain the regions regardless of the camera angle, lighting or imaging conditions. The new algorithms to be developed must be robust in low frame rate video sequences and in the presence of camera-shake, blurred images and illumination changes caused by flash-lighting such as emergency vehicle lighting, fire and flash photography.

We are working with the Met Police and other European partners to better understand the type of visual data that is available to police investigations, the problems that they face and how they are trying to solve them. With this understanding, we aim to apply contemporary

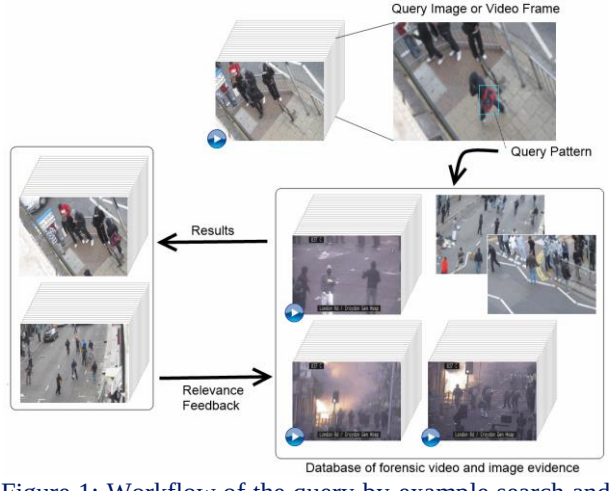

Figure 1: Workflow of the query-by-example search and retrieval system. A user draws a rectangle around a region of interest and the system searches within a database of forensic evidence for all video sequences and still images containing the region. The poor quality of the images within the database establishes the motivation for our research.

techniques in search and retrieval and forensic analysis of video sequences, and improve them to perform robustly with reliability and scalability.

Our goal is to develop a content-based information retrieval system such as illustrated in Figure 1 which provides forensic investigators with the facility to find security videos and images relevant to an investigation based on a query-by-example archetype. A database will be populated with videos and images collected as a part of an investigation, and operators will view them onscreen. When an item of interest is identified by the operator, they pause the video and draw a rectangle around a region or pattern such as a brand logo, a colour pattern on a scarf or hat, a tattoo or other distinctive marking on a person, object or clothing. The system then searches the forensic database for occurrences of the pattern and presents them to the user. A relevance feedback loop then provides the user a chance to select or remove appropriate results and the system is guided to refine the results.

2 Boosting feature correspondence using colour

In mainstream content-based image retrieval (CBIR) systems, colour can be useful, but is not absolutely discriminative. A bus is a bus regardless of its colour, and a query to a CBIR system for a bus would be expected to find buses of all colours. In such a system, colour can be treated as an attribute that may or may not be used in the search. A search for a Daffodil in a database of images of flowers may consider colour as an important search criteria as all Daffodils are a variant of yellow. In our environment of matching images from security videos, we are interested in finding and tracking a specific *instance* of an object rather than a *category* of object, and so colour can be used as a discriminating factor in matching correspondences. In a crowd scene containing two people with similarly patterned shirts, colour can be used to identify the correct shirt, for example, and the effectiveness of popular gradient-based descriptors such as SIFT [5] and SURF [6] is limited.

Our first contribution focusses on colour images where colour is a significant, albeit sometimes subtle, visual discriminator in identifying items within an image and we establish a method to improve the robustness of matching features by using colour information to boost discriminative properties of a feature descriptor. We have developed an efficient and generic extension for feature descriptors that uses colour information to improve the performance of discriminative matching of features between colour images.

2.1 Method

Using a three-step process, we create a new feature descriptor representing local key point features with colour information from the surrounding region in HSV colour space. First, a local feature detector is used to identify feature locations, and both a key point and a region are defined for each. In the case of a key point detector such as SIFT, a circular region is created with its centre at the key point co-ordinates. For region based feature detectors such as MSER, the region is approximated by an ellipse using an ellipse fitting algorithm through the region boundary points. A key point is then defined at the centre of the ellipse.

With the resulting set of key point locations and region definitions, step 2 extracts a base descriptor at each key point, describing the local texture. The base descriptor is a

standard feature descriptor that will be extended by our method to improve its discriminative capability in colour images. In step 3 we build a local colour histogram model of each region to create an extension descriptor. Using the region shape as a mask over the colour image, pixels falling within the shape are quantised into a local colour histogram representing the region. This histogram is transformed into a feature descriptor using the histogram bins as the feature dimension. Finally, the two descriptors from the base extractor and the colour extension are independently *L1*-normalized to unit length using $\|x\|_1 := \sum_{i=1}^{n} |x_i|$ and concatenated to yield a composite descriptor for each region.

2.2 Feature distance

Two features are considered to *match* if they are close to each other in feature space. A simple and common method to select a match is to calculate if the distance between the two vectors (the length of vector **c** in Figure 2A) is below a pre-defined or dynamically calculated threshold.

In designing an algorithm to extend an existing feature descriptor, consideration is made to the potential of falsely matching dissimilar features of similar colour, or perhaps worse, moving vectors in feature space closer together where neither their feature descriptor nor the colour are similar. Our goal is to produce a generic extension that can be used with any underlying base feature descriptor, and so we focus on methods to combine an *n1*-dimensional feature descriptor with an *n2*-dimensional colour histogram in such a way as to discriminate between similar features of different colours without these pitfalls.

A naïve implementation may simply extend the dimensionality of the descriptor by concatenating the feature descriptor vector and the colour-histogram into an $m = (n1 + n2)$ dimensional descriptor and consider the combined descriptors as a whole (Figure 2B). Each descriptor will be moved in m dimensional space and their relative position will be non-linear with the effect of the extension. Calculating a Euclidean distance between two extended vectors will therefore

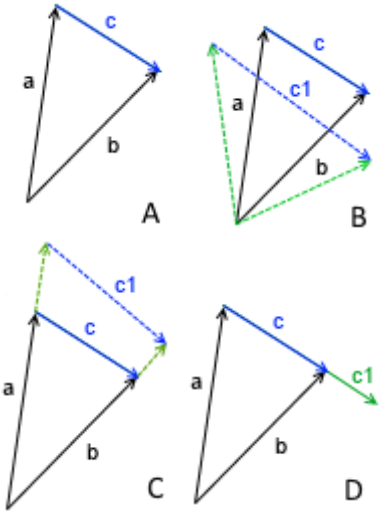

Figure 2: Means of extending vectors in feature space, shown as a 2D example. The principle extends to n-dimensions. A, vector $\mathbf{c} = \mathbf{a} + \mathbf{b}$ is drawn between the two feature vector positions, and the magnitude $\|\mathbf{c}\|$ gives the distance between a and b. B, appending a colour histogram to a feature descriptor and treating it in unity moves each feature in feature space, effecting the distance between the two descriptors. C, each vector is extended along its direction proportional to a measure of colour. If the colours are similar, each vector is extended a similar amount, but they move further apart, $\|c1\| > \|c\|$. D, our method extends the vector c between the features along its direction to yield a distance measure that doesn't affect the position of underlying features in feature space.

yield poor results. Alternatively, each feature descriptor vector can be considered independently and extended along its direction proportional to some measure of its colour. If the colours are similar, each vector is extended a similar amount, but they move further apart (Figure 2C). The distance between the extended feature descriptors will therefore be proportionally greater than the distance between the baseline descriptors.

Lowe [5] refined matching features to their closest in feature space using a *distance ratio* to determine if the closest match is a good match. The distance ratio method finds the nearest two features and divides the closest distance by the second closest. If the closest feature has another feature nearby, then the match is more likely to be incorrect. Tests in the original paper suggest that 0.8 is a reasonable threshold for this ratio, based on analysis of 40,000 keypoints, and that matches with a distance ratio greater than 0.8 should be considered less reliable as a good match. We follow this understanding in our method and use the colour information of both features to scale the distance *between* their descriptors without moving the vector in feature space (Figure 2D). In doing this, the colour detail logically moves the features apart along the vector between the descriptors. Given two feature descriptors **a** and **b**, the vector between them is defined as $\mathbf{c} = \mathbf{a} + \mathbf{b}$ and the magnitude of this vector $\|\mathbf{c}\|$ gives the distance between features. We multiply this magnitude by a value derived from the similarity of the quantised colour histograms, thus differentiating similar features by their colour.

The composite feature descriptor F is conveniently represented as a single n-dimensional vector where n is the sum of the base and extension descriptor lengths, $F = (B, E)$ and $n = len(F) = len(B) + len(E)$. In calculating the distance D between two extended descriptors, we therefore first consider a distance measure between each of the two parts independently, d_1 and d_2, and combine the results to form a distance measure D between the two composite descriptors. We use the colour difference measure as a scalar to the distance between two descriptors, which ensures that attributes of the base descriptor are preserved, such as invariance to affine scale and rotation transformations. Calculating the colour histogram using Hue and Saturation channels also maintains invariance in affine illumination transformations.

Definition

The base descriptor distance $d_1 = \|B_1 + B_2\|_2$ is a standard calculation of the Euclidean distance between the two vectors. The distance between the colour extension vectors is derived from the standard measure for the similarity of two histograms with matching bins, the Normalised Histogram Intersection, defined in [7]. Subtracting the similarity from 1 gives a dissimilarity measure, which is the distance d_2 between two extension descriptors

$$d_2 = 1 - \frac{\sum_{j=1}^{n} \min(E_{1j}, E_{2j})}{\sum_{j=1}^{n} E_{1j}} \quad (1)$$

The individual distance measures d_1 and d_2 are now combined to define the distance between the two composite descriptors. The distance measure of the colour extension is used to scale the distance measure of the base descriptor so that it discriminates between similar descriptors of different colours. Given d_2 is a normalised value in the range 0...1, the scaling multiplier becomes $1 + d_2$, thus the overall distance between two composite descriptors is $D = d_1(1 + d_2)$

Implementation

To find the closest descriptor D_c to a given descriptor D_i it is customary to use an algorithm based on Euclidean distance, such as *k*-Nearest Neighbour. We therefore perform a closest descriptor calculation in two parts. First the base descriptor's *k*-nearest neighbours are found using the standard algorithm with $k = 5$. For each of the five closest descriptors found, we perform the scaling multiplication described above and report the descriptor with the smallest resulting distance to be the closest. This is not guaranteed to be optimal, but in our tests increasing *k* to 10 does not improve the result. A common method to reduce computational complexity in a *k*-Nearest Neighbour search is to use an Approximate Nearest Neighbour algorithm (ANN) [8], which uses a randomised indexing method making the result non-deterministic but generally acceptable for most matching tasks. The proposed calculation is therefore expected to be an acceptable approximation too.

2.3 Colour-boosted RootSIFT

SIFT descriptors, which are histograms, were designed for use with Euclidean distance measures for comparison and matching [5]. However, it is well known that using Euclidean distance to compare histograms often yields inferior performance than using χ^2 or Hellinger measures. Arandjelović and Zisserman made this observation and proposed *RootSIFT* [9], which transforms the SIFT descriptor such that the Euclidean distance between two descriptors is equivalent to using the Hellinger kernel (Bhattacharyya's coefficient). RootSIFT yields a significantly more accurate result in calculating the distance between two descriptors used in feature descriptor matching.

Our extension described is also a histogram, and the combined descriptor is a concatenation of two histograms. We therefore apply the principles described in [9] to refine our method for SIFT descriptors. The base descriptor and extension descriptor are independently *L1*-normalised and each element in both descriptors are replaced with their square root value before each descriptor is independently *L2*-normalised and finally concatenated. This results in a composite 138-dimensional descriptor (using a 10-bin colour palette) which is comparable using a Euclidean distance with the accuracy of using Hellinger's kernel. For this combined descriptor, we can therefore remove the feature comparison calculation and two-step *k*-nearest neighbour algorithm and use a standard *k*-nearest neighbour implementation with Euclidean distance instead.

2.4 Experimental evaluation

We evaluate the performance of the proposed descriptors by measuring the accuracy of matching features between pairs of images. Our implementation uses a fixed colour palette for all images. In experiments, the 10-bin colour palette of Park et al. [10] has proven to work well. The definition of a feature match depends on the matching strategy that is applied [11]. Our intention is to measure the accuracy of our new composite feature descriptor and distance calculation. We therefore measure our results with a nearest neighbour matching algorithm without any threshold filtering, such as *nearest neighbour distance ratio* to discard poor matches. We use six feature detectors to find initial locations of interest. Four popular intensity based key point detectors; Harris corners [12], SIFT, SURF and BRISK [13], and two region detectors; MSER on grey scale representations and

MSCR on colour images. For each of these sets of features, we compare feature matching performance of descriptors extracted using intensity based SIFT and SURF and colour derivatives OpponentSIFT and OpponentSURF, each with and without our colour boost extension.

The key point detectors are chosen because of their popularity and widespread adoption in many tasks including object classification and image retrieval. BRISK was selected for its high performance and relevance for real-time processing and Harris corners are used frequently in tracking applications as the algorithm underpins Good Features To Track [14]. We are keen to show the universal improvements of our method and therefore also include region based detectors in our comparisons. Maximally Stable Extremal Regions (MSER) [15] is accepted to be a reliably effective and computationally efficient method of detecting feature regions in single channel images. Early work to extend MSER to multi-channel colour images was presented in [16] but did not achieve bottom up feature detection as in [17] where the author presents a derivative work specifically for maximally stable colour regions, MSCR.

Figure 3 shows two examples of matching feature descriptors from a region of interest within a query image to frame 47 (rows 1 and 2) and frame 54 (rows 3 and 4) of a test CCTV video, using a SIFT feature detector. Row 1 shows matches between SIFT descriptors extracted from the SIFT features within the region of interest in the query image, and frame 47. Row 2 shows matches between the same feature sets in each image, using our colour-boosted SIFT descriptors. Rows 3 and 4 show the same information for frame 54 where the image is severely blurred and the backpack is in a different position and orientation with respect to the camera position. The improvement in the number of positive matches to the backpack area is clearly visible in both cases, rows 2 and 4, and that there is a reduced number of incorrect matches to background features.

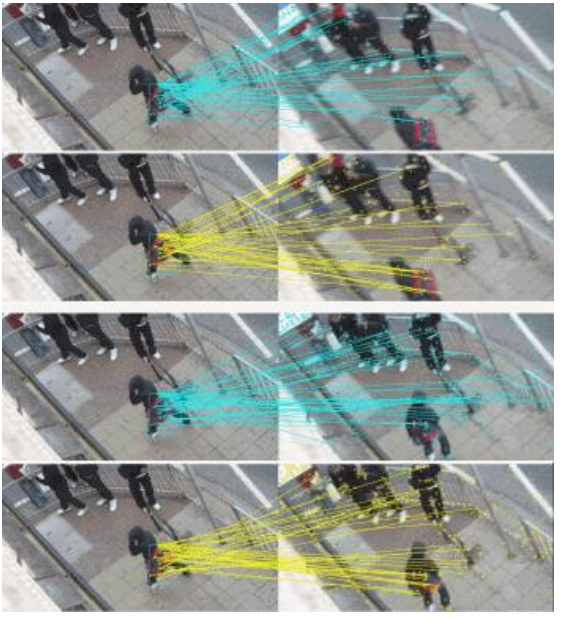

Figure 3: Query-by-example feature matching improvements using our method. The same query image and region is used in all examples. The query image is darkened to highlight the query region for display purposes only. The top two rows show matching of descriptors between the query image (left) and frame 47 (right), using the same set of SIFT features. Row 1 shows matches SIFT descriptors and Row 2 shows matches using our colour-boosted SIFT features using our method. Note the increase in correct matches to the backpack region, and also the reduced number of false matched to background features. The experiment is repeated in rows 3 and 4, with frame 54 from a test CCTV video sequence.

2.5 Conclusion

In reporting results we use the F-measure, the weighted harmonic mean of recall and precision, to quantify and compare the accuracy of our extended descriptor and distance measure with well-known descriptors. We favor neither precision nor recall over the other, and therefore use the *F1* score, defined as $\mathrm{F1} = (2 \times recall \times precision)/(recall + precision)$.

The range of accuracy that were achieved in our tests are summarised in Figure 4, showing an improvement in the best and worst performance on our dataset.

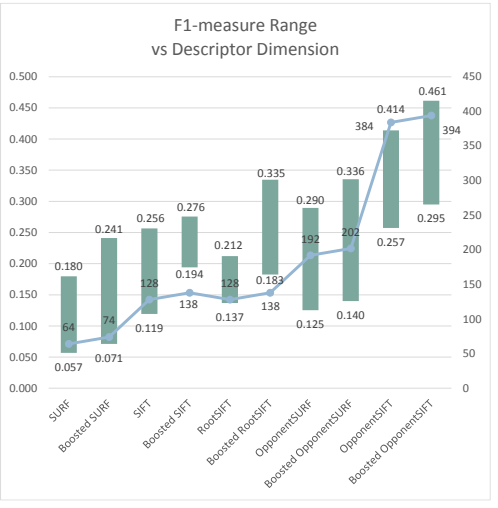

Figure 4: The range of the accuracy of different descriptors with and without our colour-boost extension shows that feature matching with the extended descriptor out-performs the baseline descriptor in our tests.

3 Learn-your-own-training-set

Our current work investigates methods to automatically discover a set of patterns with which to train a model such as a Support Vector Machine for search and retrieval. There has been little research in this area; Roth *et al.* [16] and Riemenschneider [18] have presented similar ideas and provide some prior work. **Motivation**: Using a pattern representation extracted over multiple images should enhance the model, but it is impractical for a user to define a search area containing consistent features in multiple frames of a video sequence that could make up a useful training set. **Hypothesis**: by tracking a user-selected region of interest from the query frame until it becomes occluded, goes out-of-shot or the tracking drifts too far [19], a number of representations of the region can be extracted to train a machine learning model sufficiently to determine binary classification of an object present in an image.

4 Temporal video segmentation

Bag of Visual Words [2] and derived works [3], [20], [21], [22], [23], [24], [25], [26] track features across shots within a scene. An average shot length of a modern feature film is 4-6 seconds; 100-150 frames at 25fps [27]. In contrast, continuous security footage can produce hours of video without any shot changes. Our research investigates if creating *shots of activity* can help in the forensic analysis of video for automated analysis. **Motivation**: Boundary shot detection is well documented as an important prerequisite step to automatic video content analysis [28], reducing the volume of data to process and giving subsequent algorithms video images that are focussed and manageable. **Hypothesis**: Natural breaks in the video can be identified by analysis of movement and illumination changes to find appropriate frames at which to segment the video.

References

[1] G. Edelman and J. Bijhold, "Tracking people and cars using 3D modeling and CCTV.," *Forensic science international*, vol. 202, no. 1–3, pp. 26–35, Oct. 2010.

[2] J. Sivic and A. Zisserman, "Video Google: A text retrieval approach to object matching in videos," *Proceedings of the Ninth IEEE International Conference on Computer Vision*, vol. 2, pp. 1470–1477, Oct. 2003.

[3] A. Anjulan and N. Canagarajah, "A Unified Framework for Object Retrieval and Mining," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 19, no. 1, pp. 63–76, Jan. 2009.

[4] S. O'Hara and B. A. Draper, "Introduction to the Bag of Features Paradigm for Image Classification and Retrieval," *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'11)*, Jan. 2011.

[5] D. G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, Nov. 2004.

[6] H. Bay, T. Tuytelaars, and L. Van Gool, "SURF: Speeded Up Robust Features," in *European Conference on Computer Vision (ECCV'06)*, 2006, pp. 404–417.

[7] M. Swain and D. Ballard, "Color indexing," *International Journal of Computer Vision*, vol. 7, no. 1, pp. 11–32, 1991.

[8] P. Indyk and R. Motwani, "Approximate nearest neighbors," in *Proceedings of the thirtieth annual ACM symposium on Theory of computing - STOC '98*, 1998, pp. 604–613.

[9] R. Arandjelović and A. Zisserman, "Three things everyone should know to improve object retrieval," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 2911–2918.

[10] U. Park, A. K. Jain, I. Kitahara, K. Kogure, and N. Hagita, "ViSE: Visual Search Engine Using Multiple Networked Cameras," in *18th International Conference on Pattern Recognition (ICPR'06)*, 2006, vol. 3, pp. 1204–1207.

[11] K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 10, pp. 1615–1630, 2005.

[12] C. Harris and M. Stephens, "A Combined Corner and Edge Detector," in *Procedings of the Alvey Vision Conference 1988*, 1988, pp. 23.1–23.6.

[13] S. Leutenegger, M. Chli, and R. Y. Siegwart, "BRISK: Binary Robust invariant scalable keypoints," in *2011 International Conference on Computer Vision*, 2011, pp. 2548–2555.

[14] J. Shi and C. Tomasi, "Good features to track," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition CVPR-94*, 1994, pp. 593–600.

[15] P.-E. Forssén and D. G. Lowe, "Shape Descriptors for Maximally Stable Extremal Regions," in *IEEE 11th International Conference on Computer Vision (ICCV) 2007*, 2007, pp. 1–8.

[16] P. M. Roth, M. Donoser, and H. Bischof, "Tracking for learning an object representation from unlabeled data," *Proceedings of the Computer Vision Winter Workshop (CVWW)*, pp. 46–51, 2006.

[17] P.-E. Forssén, "Maximally Stable Colour Regions for Recognition and Matching," in *Computer Vision and Pattern Recognition (CVPR) 2007*, 2007, pp. 1–8.

[18] H. Riemenschneider, "Robust Online Object Learning and Recognition by MSER Tracking," *Proc. 13th Computer Vision Winter Workshop (CVWW)*, 2007.

[19] I. Matthews, T. Ishikawa, and S. Baker, "The template update problem," *IEEE transactions on pattern analysis and machine intelligence*, vol. 26, no. 6, pp. 810–815, 2004.

[20] J. Sivic, F. Schaffalitzky, and A. Zisserman, "Object Level Grouping for Video Shots," *International Journal of Computer Vision*, vol. 67, no. 2, pp. 189–210, Jan. 2006.

[21] J. Sivic, F. Schaffalitzky, and A. Zisserman, "Efficient object retrieval from videos," *Proc of the 12th European Signal Processing Conference EUSIPCO 04 Vienna Austria*, pp. 36–39, Sep. 2004.

[22] J. Sivic and A. Zisserman, "Video data mining using con .gurations of viewpoint invariant regions," in *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.*, 2004, vol. 1, pp. 488–495.

[23] J. Sivic and A. Zisserman, "Efficient visual search of videos cast as text retrieval," *IEEE transactions on pattern analysis and machine intelligence*, vol. 31, no. 4, pp. 591–606, 2009.

[24] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman, "Object retrieval with large vocabularies and fast spatial matching," *2007 IEEE Conference on Computer Vision and Pattern Recognition*, 2007.

[25] Y.-G. Jiang, C.-W. Ngo, and J. Yang, "Towards optimal bag-of-features for object categorization and semantic video retrieval," *Proceedings of the 6th ACM international conference on Image and video retrieval - CIVR '07*, pp. 494–501, 2007.

[26] R. Shekhar and C. V. Jawahar, "Word Image Retrieval Using Bag of Visual Words," in *2012 10th IAPR International Workshop on Document Analysis Systems*, 2012, pp. 297–301.

[27] D. Bordwell, *The Way Hollywood Tells It: Story and Style in Modern Movies*. University of California Press, 2006, p. 300.

[28] J. Y. J. Yuan, H. W. H. Wang, L. X. L. Xiao, W. Z. W. Zheng, J. L. J. Li, F. L. F. Lin, and B. Z. B. Zhang, "A Formal Study of Shot Boundary Detection," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 17, 2007.