

## HYBRID CODING OF VISUAL CONTENT AND LOCAL IMAGE FEATURES

Luca Baroffio, Matteo Cesana, Alessandro Redondi, Marco Tagliasacchi, Stefano Tubaro

Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano

### ABSTRACT

Distributed visual analysis applications, such as mobile visual search or Visual Sensor Networks (VSNs) require the transmission of visual content on a bandwidth-limited network, from a peripheral node to a processing unit. Traditionally, a “Compress-Then-Analyze” approach has been pursued, in which sensing nodes acquire and encode the pixel-level representation of the visual content, that is subsequently transmitted to a sink node in order to be processed. This approach might not represent the most effective solution, since several analysis applications leverage a compact representation of the content, thus resulting in an inefficient usage of network resources. Furthermore, coding artifacts might significantly impact the accuracy of the visual task at hand. To tackle such limitations, an orthogonal approach named “Analyze-Then-Compress” has been proposed [1]. According to such a paradigm, sensing nodes are responsible for the extraction of visual features, that are encoded and transmitted to a sink node for further processing. In spite of improved task efficiency, such paradigm implies the central processing node not being able to reconstruct a pixel-level representation of the visual content. In this paper we propose an effective compromise between the two paradigms, namely “Hybrid-Analyze-Then-Compress” (HATC) that aims at jointly encoding visual content and local image features. Furthermore, we show how a target tradeoff between image quality and task accuracy might be achieved by accurately allocating the bitrate to either visual content or local features.

**Index Terms**— Local features, BRISK, Image compression, Predictive coding

### 1. INTRODUCTION

In the last few years, local features have been effectively exploited in a number of visual analysis tasks such as augmented reality, object recognition, content based retrieval, image registration, etc. They provide a robust yet concise representation of an image patch that is invariant to local and global transformation such as illumination and viewpoint changes. The traditional pipeline for the extraction of local image feature consists of two main stages: i) a keypoint detector, that aims at identifying salient points within an image and ii) a keypoint descriptor that captures the local information of the image patch surrounding each keypoint. Traditional algorithms for keypoint description, such as SIFT [2] and SURF [3], assign to each salient point a description by means of a set of real-valued elements, capturing local information based on intensity gradient. More recently, a novel class of algorithms, namely binary descriptors, has emerged as an effective, yet computationally efficient, alternative to

SIFT and SURF. Such features usually rely on smoothed pixel intensities and not on local intensity gradients, vastly improving the computational efficiency. The BRIEF [4] descriptor consists of a set of binary values, each obtained by comparing the smoothed intensity of two pixels, randomly sampled around a keypoint. BRISK [5], ORB [6] and FREAK [7] refine the process, introducing ad-hoc designed spatial patterns of pixels to be compared and achieving rotation-invariance. More recently, BAMBOO [8][9] exploits a pairwise boosting algorithm to build a discriminative pattern of pairwise pixel intensity comparisons.

Local features represent a key component of many distributed visual analysis applications such as Mobile Visual Search, augmented reality, and Visual Sensor Networks applications. Traditionally, such tasks have been tackled according to a *Compress-Then-Analyze* (CTA) approach, in which sensing nodes acquire the content, encode it resorting to picture or video coding primitives, e.g. JPEG or H.264/AVC, and transmit it to a central server that extracts local features and performs a given visual analysis task. According to CTA, the pixel-level representation of the acquired visual content is actually sent to the sink node. A number of applications rely on compact representations of the content, in the form of local or global features. In this context, CTA might not be the most efficient approach, since unnecessary and possibly redundant information is sent on the network. Furthermore, the central processing node receives and exploits a lossy version of the originally acquired visual content. Artifacts introduced by coding algorithms may affect the accuracy of several applications [1]. Several works in the literature aim at adapting both image [10] and video [11, 12] compression architectures so that the quality local features is preserved.

An alternative paradigm, namely *Analyze-Then-Compress* (ATC), has been introduced in [1]. Such an approach aims at tackling the limitations posed by CTA. According to ATC, the sensing nodes acquire the visual content, extract information in the form of local or global features, that are encoded and transmitted to a sink node that performs visual analysis based on such features. Such paradigm moves part of the computational complexity from the central unit to the sensing nodes. To this end, efficient algorithms for visual feature extraction [9, 13] and coding architectures tailored to global and local visual features [14, 15, 16, 17] have been proposed. The task efficiency is improved, since only relevant information is actually transmitted over the network. Still, the sink node is not able to reconstruct the original pixel-level representation of the visual content.

In this paper we propose a novel hybrid approach to distributed visual analysis tasks aimed at overcoming the limitations of both ATC and CTA. *Hybrid-Analyze-Then-Compress* (HATC) represents an efficient solution for the joint coding of both pixel-level and local feature-level representations. Furthermore, the allocation of the bit budget to either visual content or image feature is thoroughly investigated.

Moulin et al. [18] addressed the problem of jointly encoding

The project GreenEyes acknowledges the financial support of the Future and Emerging Technologies (FET) programme within the Seventh Framework Programme for Research of the European Commission, under FET-Open grant number:296676.

pixel-level content and global image features such as either Bag-of-Words histograms or integral channel features in the context of scene classification or pedestrian detection, respectively. Differently, we focus on the joint encoding of visual content and local image features, typically consisting of sets of salient points, along with their descriptors.

The rest of this paper is organized as follows: Section 2 introduces the problem, defining tools and objectives, Section 3 describes the proposed paradigm, Section 4 is devoted to experimental evaluation. Finally, Section 5 draws conclusions and discusses future work.

## 2. PROBLEM STATEMENT

Let  $\mathcal{I}$  denote an image that is acquired by a sensing node. Such image is processed in order to extract a set of features  $\mathcal{D}$ . To this end, a detector is applied to the image in order to identify interest points. The number of detected keypoints  $M = |\mathcal{D}|$  depends on both the image content and on the type and parameters of the adopted detector. Then, a keypoint descriptor is computed starting from the orientation-compensated patch surrounding each interest point. Hence,  $d_m \in \mathcal{D}$  is a local feature, that consists of two components: i) a 4-dimensional vector  $\mathbf{p}_m = [x_m, y_m, \sigma_m, \theta_m]^T$ , indicating the position  $(x_m, y_m)$ , the scale  $\sigma_m$  of the detected keypoint, and the orientation angle  $\theta_m$  of the image patch; ii) a  $D$ -dimensional vector  $\mathbf{d}_m$ , which represents the descriptor associated to the keypoint  $\mathbf{c}_m$ . According to *Analyze-Then-Compress*, the set of features  $\mathcal{D}$  is encoded and transmitted to a sink node for further analysis. On the other hand, *Compress-Then-Analyze* would require the acquired image  $\mathcal{I}$  to be encoded and transmitted to a central unit where it is analyzed. In details, the sink node receives the bitstream and reconstructs a lossy version of the original image  $\tilde{\mathcal{I}}$ . Then, similarly to the case of ATC, a set of local descriptors is extracted and exploited to perform a given visual analysis task. However, the image coding process introduces artifacts that may affect the extraction of local features and, as a consequence, the task accuracy.

We propose an alternative approach, namely *Hybrid-Analyze-Then-Compress*, that aims at efficiently coding both pixel-domain and feature-domain representations of the visual content. In particular, according to such paradigm, the decoder is capable of reconstructing both a lossy representation of the original image  $\tilde{\mathcal{I}}$  (encoded with  $R_{\tilde{\mathcal{I}}}$  bits) and a subset of the original features  $\mathcal{D}_{\text{HATC}}$  (encoded with  $R_{\mathcal{D}_{\text{HATC}}}$  bits), thus requiring  $R_{\text{HATC}} = R_{\tilde{\mathcal{I}}} + R_{\mathcal{D}_{\text{HATC}}}$  in total.

The HATC approach is generally applicable to any kind of local feature. In this paper, we focus on the case in which binary descriptors are used, i.e.,  $\mathbf{d}_m \in \{0, 1\}^D$ . Each descriptor element is a bit, representing the result of a pairwise comparison of smoothed pixel intensities sampled from an ad-hoc designed pattern around a given interest point. In particular, we consider BRISK [5] binary features.

## 3. HATC CODING ARCHITECTURE

Figure 1 illustrates the pipeline of the HATC coding architecture. As regards the coding of the pixel-level representation of the visual content, HATC is equivalent to the CTA approach. That is, the acquired image is encoded and sent to the sink node. Here, the bitstream is decoded and a lossy representation of the image  $\tilde{\mathcal{I}}$  is reconstructed. CTA would run a detector and a descriptor algorithm on  $\tilde{\mathcal{I}}$ , obtaining visual features whose effectiveness is possibly impaired by the image coding artifacts. The key idea behind HATC is to add an enhancement layer that allows the central processing node to

reconstruct a subset  $\mathcal{D}_{\text{HATC}}$  of the original local descriptors  $\mathcal{D}$ . Such approach allows for the refinement of an arbitrarily-sized subset of features extracted from lossy pixel-level content, yielding a tradeoff between bitrate and task accuracy. The higher the number  $Z$  of features that are refined, the higher the resulting bitrate and the higher the accuracy of the visual analysis task to be performed.

To construct the feature enhancement layer, the sensing node extracts a set of interest points  $\mathcal{K}$  from the acquired image  $\mathcal{I}$ . The sensing node computes the sets of descriptors  $\mathcal{D}$  and  $\tilde{\mathcal{D}}$  from the original image  $\mathcal{I}$  and  $\tilde{\mathcal{I}}$ , respectively. Descriptors are computed in correspondence to the locations defined by the set  $\mathcal{K}$ . Finally, a subset  $\mathcal{D}_{\text{HATC}}$  of the set of original descriptors  $\mathcal{D}$  is differentially encoded with respect to the set of lossy descriptors  $\tilde{\mathcal{D}}$ .

At the central processing node, a lossy representation  $\tilde{\mathcal{I}}$  of the original image is decoded, along with the set of keypoint locations  $\mathcal{K}$ . The set of descriptors  $\tilde{\mathcal{D}}$  is computed exploiting the lossy coded image  $\tilde{\mathcal{I}}$ , at the locations defined by  $\mathcal{K}$ . Finally, the bitstream related to the enhancement layer  $\mathcal{D}^*$  is decoded and exploited in order to reconstruct the subset  $\mathcal{D}_{\text{HATC}}$  of the original descriptors  $\mathcal{D}$ .

The HATC paradigm requires three main components to be encoded and transmitted to the central node:

- $\mathcal{I}^*$ , i.e., the bitstream needed to reconstruct a lossy representation  $\tilde{\mathcal{I}}$  of the original image  $\mathcal{I}$ ;
- $\mathcal{K}^*$ , i.e., the bitstream needed to reconstruct the location of the keypoints extracted from the original image  $\mathcal{I}$ ;
- $\mathcal{D}_{\text{HATC}}^*$ , i.e., the bitstream needed to reconstruct the feature enhancement layer  $\mathcal{D}_{\text{HATC}}$ .

In a summary, HATC offers advantages with respect to both ATC and CTA. First, differently from ATC, the central unit is capable of reconstructing the pixel-level visual content. Second, differently from CTA, HATC allows the sink node to operate on high quality visual features, yielding a higher task accuracy.

### 3.1. Differential coding of binary local features

For HATC to be competitive with other approaches, an effective ad-hoc coding architecture has to be developed. Consider the sets of descriptors  $\mathcal{D}$  and  $\tilde{\mathcal{D}}$ , extracted from an input image  $\mathcal{I}$  and its lossy counterpart  $\tilde{\mathcal{I}}$ , respectively. The proposed differential coding architecture aims at efficiently encoding the descriptors  $\mathcal{D}_{\text{HATC}}$ , exploiting  $\tilde{\mathcal{D}}$  as a predictor. The key tenet behind HATC is that the two sets of descriptors, extracted in correspondence of a common set of interest point locations, are correlated. In a sense, such a scenario is similar to that of features extracted from contiguous frames of a video sequence. In that case, inter-frame predictive coding can be exploited to improve coding efficiency, reducing the output bitrate [14, 15, 19].

In the case of HATC, given a binary descriptor  $\mathbf{d}_m \in \mathcal{D}$  and its counterpart  $\tilde{\mathbf{d}}_m \in \tilde{\mathcal{D}}$  extracted from the original and the decoded images, respectively, the prediction residual can be computed as

$$\mathbf{c}_m = \mathbf{d}_m \oplus \tilde{\mathbf{d}}_m, \quad (1)$$

that is, the bitwise XOR between  $\mathbf{d}_m$  and  $\tilde{\mathbf{d}}_m$ .

In binary descriptors, each element represents the binary outcome of a pairwise comparison between smoothed pixel intensities. Hence, the dexels (descriptor elements) are potentially statistically dependent, and so are the elements of the prediction residual  $\mathbf{c}_m$ . In this context, it is possible to model the prediction residual as a binary source with memory. Let  $\pi_j, j \in [1, D]$  represent the  $j$ -th

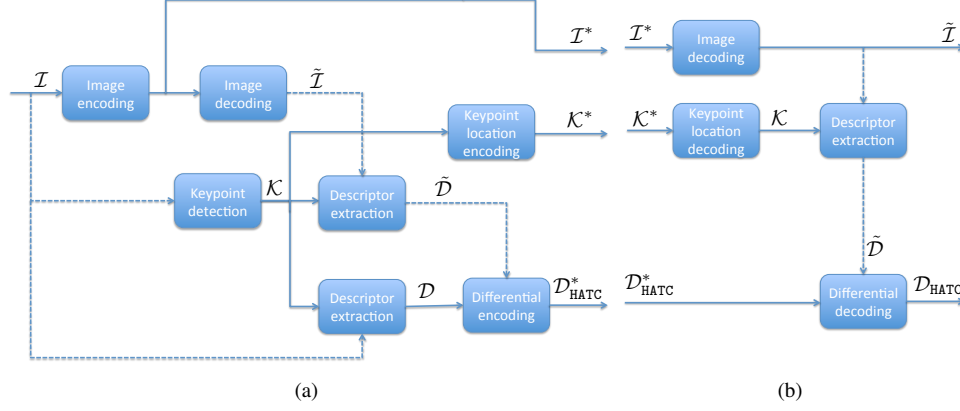


Fig. 1. Block diagram of a) HATC joint feature-image encoder; b) HATC joint feature-image decoder.

element of a prediction residual, where  $D$  is the dimension of such a descriptor. The entropy of such an element can be computed as

$$H(\pi_j) = -p_j(0) \log_2(p_j(0)) - p_j(1) \log_2(p_j(1)), \quad (2)$$

where  $p_j(0)$  and  $p_j(1)$  are the probability of  $\pi_j = 0$  and  $\pi_j = 1$ , respectively. Similarly, the conditional entropy of element  $\pi_{j_1}$  given element  $\pi_{j_2}$  can be computed as

$$H(\pi_{j_1} | \pi_{j_2}) = \sum_{x \in \{0,1\}, y \in \{0,1\}} p_{j_1, j_2}(x, y) \log_2 \frac{p_{j_2}(y)}{p_{j_1, j_2}(x, y)}, \quad (3)$$

with  $j_1, j_2 \in [1, D]$ . Let  $\tilde{\pi}_j, j = 1, \dots, D$ , denote a permutation of the prediction residual elements, indicating the sequential order used to encode a descriptor. The average code length needed to encode a descriptor is lower bounded by

$$R = \sum_{j=1}^P H(\tilde{\pi}_j | \tilde{\pi}_{j-1}, \dots, \tilde{\pi}_1). \quad (4)$$

In order to maximize the coding efficiency, we aim at finding the permutation of elements  $\tilde{\pi}_1, \dots, \tilde{\pi}_D$  that minimizes such a lower bound. For the sake of simplicity, we model the source as a first-order Markov source. That is, we impose  $H(\tilde{\pi}_j | \tilde{\pi}_{j-1}, \dots, \tilde{\pi}_1) = H(\tilde{\pi}_j | \tilde{\pi}_{j-1})$ . Then, we adopt the following greedy strategy to reorder the elements of the prediction residual:

$$\tilde{\pi}_j = \begin{cases} \arg \min_{\pi_j} H(\pi_j) & j = 1 \\ \arg \min_{\pi_j} H(\pi_j | \tilde{\pi}_{j-1}) & j \in [2, D] \end{cases} \quad (5)$$

Note that such optimal ordering is computed offline, thanks to a training phase, and shared between both the encoder and the decoder.

### 3.2. Coding of keypoint locations

Consider an  $N_x \times N_y$  image  $\mathcal{I}$ . The coordinates of each keypoint  $\mathbf{c}_m \in \mathcal{K}$  (at quarter-pel accuracy) are encoded using  $R_{\mathbf{c}_m} = M_n(\log_2 4N_x + \log_2 4N_y + S)$  bits, where  $S$  is the number of bits used to encode the scale parameter. Higher coding efficiency is achievable implementing ad-hoc lossless or lossy coding schemes to compress the coordinates of the keypoints [20][21].

## 4. EXPERIMENTS

The effectiveness of the proposed paradigm has been evaluated and compared with that of both *Compress-Then-Analyze* and *Analyze-Then-Compress*, with respect to a content-based image retrieval application.

### 4.1. Datasets

We exploit the publicly available *Zurich building dataset* (ZuBuD) [22] in order to evaluate the performance of HATC. Such a dataset consists of 1005 pictures representing 201 different Zurich buildings (5 different views for each object). A test set composed of 115 image queries, each one capturing a different building, is also provided. Database and query images have heterogeneous resolutions and imaging conditions. As regards the training phase, 1000 images have been randomly sampled from the MIRFLICKR [23] dataset and they have been exploited to compute the coding-wise optimal dixel order and the associated coding probabilities, as illustrated in Section 3.

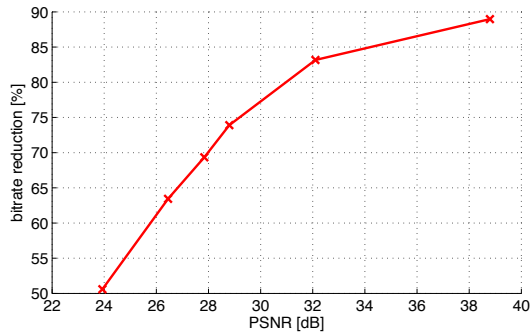
### 4.2. Methods

We compared the performance of the following paradigms:

- *Compress-Then-Analyze* (CTA): each query picture is encoded resorting to JPEG. Subsequently, BRISK local features are extracted from the lossy compressed image and exploited for the retrieval pipeline;
- *Analyze-Then-Compress* (ATC): each query picture is processed in order to extract a set of BRISK features, that are encoded resorting to the architecture proposed in [16] and exploited for the retrieval pipeline;
- *Hybrid-Analyze-Then-Compress* (HATC): a local feature enhancement layer, composed by a subset of the BRISK feature extracted from the uncompressed image, is generated and differentially encoded according to the procedure presented in Section 3. Such features are exploited for the retrieval pipeline.

### 4.3. Parameter settings

As for CTA, we define a set of possible values for the JPEG quality factor  $Q = \{5, 10, 15, 20, 50, 70\}$  in order to generate



**Fig. 2.** Feature coding efficiency as a function of the distortion (PSNR) between the original and the lossy pixel-level visual content.

a rate-accuracy curve. As to ATC, a similar rate-accuracy curve is obtained by imposing different BRISK detection thresholds  $t_{BRISK} = \{70, 75, 80, 85, 90, 95, 100, 105\}$ . Finally, as to HATC, for each JPEG quality factor, a rate-accuracy curve is obtained by setting the number  $Z = \{25, 50, 100, 150\}$  of features to be refined resorting to a feature enhancement layer, as reported in Section 3.

#### 4.4. Evaluation metrics

We evaluate the performance in terms of rate-accuracy curves. In particular, the accuracy of the task is evaluated according to the *Mean Average Precision* (MAP) measure. Given an input query image  $\mathcal{I}_q$ , it is possible to define the *Average Precision* as

$$AP_q = \frac{\sum_{k=1}^Z P_q(k)r_q(k)}{R_q}, \quad (6)$$

where  $P_q(k)$  is the precision (i.e., the fraction of relevant documents retrieved) considering the top- $k$  results in the ranked list of database images;  $r_q(k)$  is an indicator function, which is equal to 1 if the item at rank  $k$  is relevant for the query, and zero otherwise;  $R_q$  is the total number of relevant document for query  $\mathcal{I}_q$  and  $Z$  is the total number of documents in the list. The overall *Mean Average Precision* for the whole set of query images is computed as

$$MAP = \frac{\sum_{q=1}^Q AP_q}{Q}, \quad (7)$$

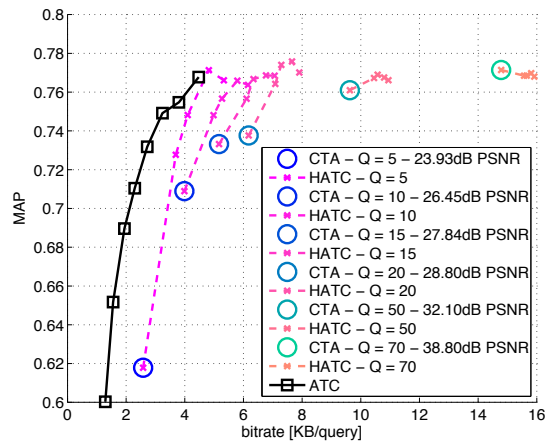
where  $Q$  is the total number of queries.

The quality of a JPEG coded image is evaluated according to its PSNR with respect to the uncompressed image.

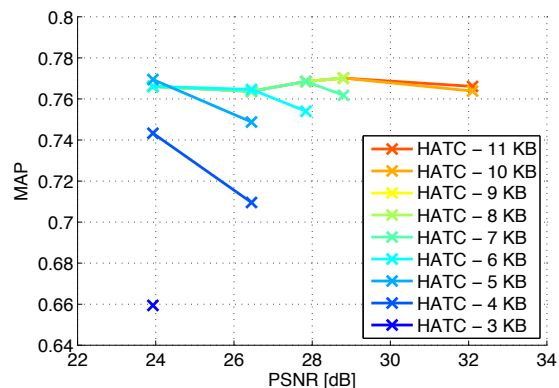
#### 4.5. Results

Figure 2 shows the feature coding efficiency achieved by the differential encoding module (see Figure 1) as a function of the distortion (PSNR) between the original image and the lossy one reconstructed resorting to CTA. The lower the distortion (the higher the PSNR), the more effective the HATC feature coding architecture. Nonetheless, high PSNRs correspond to low distortion values, and thus the accuracy increment yield by HATC is smaller.

Figure 3 compares the rate-accuracy performance of the three approaches. For example, when 4 KB/query are allocated, CTA achieves a MAP equal to 0.71. This value increases to 0.75 when using HATC, trading-off accuracy for visual quality (which decreases



**Fig. 3.** Rate-accuracy curves comparing the performance of ATC, CTA and HATC.



**Fig. 4.** Tradeoff between pixel-level distortion (PSNR) and visual analysis task accuracy (MAP) obtained resorting to the HATC architecture. Each curve refers to a target bitrate budget.

from 26.4dB to 23.9dB). ATC achieves a slightly higher MAP (0.76), but the pixel-domain content is not available at the decoder. A similar analysis can be performed for different target bitrate budgets. Figure 4 shows the MAP-PSNR trade-offs that are achievable when targeting a given bitrate. When the available bitrate is equal to 3KB per query, a single working point corresponding to 0.66 MAP @ 24dB PSNR is achievable. At higher target bitrates (e.g. 4-7 KB/query), it is possible to select a trade-off between MAP and PSNR by accurately allocating the available bitrate to either the pixel-level or the feature-level representations.

## 5. CONCLUSIONS

In this paper we propose *Hybrid-Analyze-Then-Compress*, an effective paradigm tailored to distributed visual analysis tasks. Such model exploits a joint pixel- and local feature-level coding architecture, leading to significant bitrate savings. Future work will aim at improving the coding efficiency of both the keypoint location and the descriptor enhancement layer modules and at extending the approach to different classes of local features (e.g. SIFT, SURF descriptors).

## 6. REFERENCES

- [1] L. Baroffio, M. Cesana, A. Redondi, and M. Tagliasacchi, "Compress-then-analyze vs. analyze-then-compress: Two paradigms for image analysis in visual sensor networks," in *IEEE International Workshop on Multimedia Signal Processing (MMSP) 2013*, Pula, Italy, September 2013.
- [2] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.
- [3] H. Bay, T. Tuytelaars, and L. J. Van Gool, "Surf: Speeded up robust features," in *ECCV (1)*, 2006, pp. 404–417.
- [4] M. Calonder, V. Lepetit, C. Strecha, and P. Fua, "Brief: Binary robust independent elementary features," in *ECCV (4)*, 2010, pp. 778–792.
- [5] S. Leutenegger, M. Chli, and R. Siegwart, "Brisk: Binary robust invariant scalable keypoints," in *ICCV*, 2011, pp. 2548–2555.
- [6] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "Orb: An efficient alternative to sift or surf," in *Computer Vision (ICCV), 2011 IEEE International Conference on*, Nov 2011, pp. 2564–2571.
- [7] A. Alahi, R. Ortiz, and P. Vanderghenst, "Freak: Fast retina keypoint," in *CVPR*, 2012, pp. 510–517.
- [8] L. Baroffio, M. Cesana, A. Redondi, and M. Tagliasacchi, "Binary local descriptors based on robust hashing," in *IEEE International Workshop on Multimedia Signal Processing (MMSP) 2013*, Pula, Italy, September 2013.
- [9] L. Baroffio, M. Cesana, A. Redondi, and M. Tagliasacchi, "Bamboo: a fast descriptor based on asymmetric pairwise boosting," in *submitted to IEEE International Conference on Image Processing 2014*, Paris, France, October 2014.
- [10] Jianshu Chao and E. Steinbach, "Preserving sift features in jpeg-encoded images," in *Image Processing (ICIP), 2011 18th IEEE International Conference on*, Sept 2011, pp. 301–304.
- [11] D. Agrafiotis, D.R. Bull, Nishan Canagarajah, and N. Kamnoonwatana, "Multiple priority region of interest coding with h.264," in *Image Processing, 2006 IEEE International Conference on*, Oct 2006, pp. 53–56.
- [12] Jui-Chiu Chiang, Cheng-Sheng Hsieh, G. Chang, Fan-Di Jou, and Wen-Nung Lie, "Region-of-interest based rate control scheme with flexible quality on demand," in *Multimedia and Expo (ICME), 2010 IEEE International Conference on*, July 2010, pp. 238–242.
- [13] T. Trzcinski, M. Christoudias, V. Lepetit, and P. Fua, "Boosting Binary Keypoint Descriptors," in *Computer Vision and Pattern Recognition*, 2013.
- [14] L. Baroffio, M. Cesana, A. Redondi, M. Tagliasacchi, and S. Tubaro, "Coding visual features extracted from video sequences," *Image Processing, IEEE Transactions on*, vol. 23, no. 5, pp. 2262–2276, May 2014.
- [15] L. Baroffio, J. Ascenso, M. Cesana, A. Redondi, S. Tubaro, and M. Tagliasacchi, "Coding binary local features extracted from video sequences," in *IEEE International Conference on Image Processing*, Paris, France, 2014.
- [16] A. Redondi, L. Baroffio, J. Ascenso, M. Cesana, and M. Tagliasacchi, "Rate-accuracy optimization of binary descriptors," in *20th IEEE International Conference on Image Processing*, Melbourne, Australia, September 2013.
- [17] M. Makar, S. S. Tsai, V. Chandrasekhar, D. Chen, and B. Girod, "Interframe coding of canonical patches for low bit-rate mobile augmented reality," *International Journal of Semantic Computing*, vol. 7, no. 1, pp. 5–24, 2013.
- [18] Scott Chen and Pierre Moulin, "A two-part predictive coder for multitask signal compression," in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Florence, Italy, May 2014.
- [19] L. Baroffio, A. Redondi, M. Cesana, S. Tubaro, and M. Tagliasacchi, "Coding video sequences of visual features," in *20th IEEE International Conference on Image Processing*, Melbourne, Australia, September 2013.
- [20] Sam S. Tsai, David Chen, Gabriel Takacs, Vijay Chandrasekhar, Jatinder P. Singh, and Bernd Girod, "Location coding for mobile image retrieval," in *Proceedings of the 5th International ICST Mobile Multimedia Communications Conference, ICST*, Brussels, Belgium, Belgium, 2009, Mobimedia '09, pp. 8:1–8:7, ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering).
- [21] Sam S. Tsai, David Chen, Gabriel Takacs, Vijay Chandrasekhar, Mina Makar, Radek Grzeszczuk, and Bernd Girod, "Improved coding for image feature location information," *Proc. Applications of Digital Image Processing XXXV, Proc. SPIE vol. 8499*.
- [22] H. Shao, T. Svoboda, and L. Van Gool, "ZuBuD — Zurich buildings database for image based recognition," Tech. Rep. 260, Computer Vision Laboratory, Swiss Federal Institute of Technology, March 2003.
- [23] Mark J. Huiskes and Michael S. Lew, "The mir flickr retrieval evaluation," in *MIR '08: Proceedings of the 2008 ACM International Conference on Multimedia Information Retrieval*, New York, NY, USA, 2008, ACM.