A Prediction-Based Approach for Features Aggregation in Visual Sensor Networks

Alessandro Enrico Redondi, Matteo Cesana, Luigi Fratta, Antonio Capone, Flaminio Borgonovo

Dipartimento di Elettronica, Informazione e Bioingegneria
Politecnico di Milano

Abstract

Visual Sensor Networks (VSNs) constitute a key technology for the implementation of several visual analysis tasks. Recent studies have demonstrated that such tasks can be efficiently performed following an operative paradigm where cameras transmit to a central controller local image features, rather than pixel-domain images. Furthermore, features from multiple camera views may be efficiently aggregated exploiting the spatial redundancy between overlapping views. In this paper we propose a routing protocol designed for supporting aggregation of image features in a VSN. First, we identify a predictor able to estimate the efficiency of local features aggregation between different cameras in a VSN. The proposed predictor is chosen so as to minimize the prediction error while keeping the network overhead cost low. Then, we harmonically integrate the proposed predictor in the Routing Protocol for Low-Power and Lossy Networks (RPL) in order to support the task of in-network feature aggregation. We propose a RPL objective function that takes into account the predicted aggregation efficiency and build the routes from the camera nodes to a central controller so that either energy consumption or used network bandwidth is minimized. Extensive experimental results confirm that the proposed approach can be used to increase the efficiency of VSNs.

Keywords: Visual Sensor Networks, Multi-View Features Compression, RPL

1. Introduction

In the last few years, Visual Sensor Networks (VSNs) have emerged as an important class of distributed networked systems. Composed of many low-cost, battery-operated wireless camera sensors with the ability of acquiring, processing and transmitting visual data, VSNs extend the capabilities of traditional Wireless Sensor Networks (WSNs), and they are expected to play a major role in the evolution of the Internet-of-Things (IoT) paradigm. Being able to gather, process and analyse visual data, VSNs constitute a key technology to implement several monitoring and analysis applications in remote and inaccessible scenarios. As an example, in the field of environmental monitoring, battery-operated wireless cameras can be employed to keep track of the changes in the characteristics of the vegetation or other environmental features over time [1]; in the fields of ecology, conservation biology and wildlife monitoring, VSNs are deployed in remote locations and coupled with additional sensors (e.g., motion or infrared) to capture and possibly analyze images of wild animals [2]. VSNs find application also in the context of Smart Cities: although it is reasonable to assume that most of the cameras deployed in urban environments do not run on batteries and have high-bandwidth communication capabilities, the possibility to quickly deploy many wireless-capable battery-operated cameras opens the way to the implementation of several services (traffic and infrastructure monitoring, vacant parking lot detection, etc.) in a much more flexible way than what achievable with traditional wired deployments. VSNs are also studied coupled with vehicular mobility to build up powerful vehicular sensor networks to support safety and smart city applications [3, 4]. VSNs are particularly stimulating from the research point of view as they pose additional challenges compared to traditional WSNs. Such challenges come from the struggle between applications requirements and technology constraints: on the one hand, applications based on visual data generally require intense processing and high bandwidth availability. On the other hand, VSNs are characterized by tight energy, processing and bandwidth constraints, thus calling for advanced solutions in the areas of data compression, processing and networking.

In VSNs, multiple camera sensors are generally deployed in the same area, and it is very likely that their fields of view (FoVs) overlap. This is done either for increasing the robustness of monitoring (e.g, ensuring visual coverage even in case of camera failures) or the monitoring accuracy (avoiding occlusions,

fusing information from multiple points of view, etc.). As a result, visual data streams from multiple cameras may exhibit high degrees of correlation: considering the energy and bandwidth constraints of VSNs, it is imperative to find efficient ways to remove such redundancy before data transmission. When the data transmitted from camera nodes is composed of images and video, it is possible to rely on recent advances in the fields of multi-view coding (MVC) [5] and distributed video coding (DVC) [6] in order to compress and aggregate correlated image data in networked scenarios. Such solutions generally aim at maximizing the overall compression efficiency following a two steps approach: first, the correlation existing between different cameras is predicted using either geometric [7] or content-dependent information [8]; then, based on this prediction, routing in the network is optimized so as to maximize different performance metrics (e.g., lifetime [9, 10] or quality-of-service [11]).

However, recent works in the field of VSNs and embedded vision systems have demonstrated that pixel-domain (images and videos) data transmission from camera nodes to a remote server performs poorly in bandwidth- and energy-constrained scenarios [12]. At the same time, an alternative approach is gaining popularity: instead of acquiring, compressing and transmitting *images* to a central server for further analysis, camera nodes may extract, compress and transmit *visual features* to the server. Such visual features have a series of favorable characteristics when used in constrained scenarios: first, they summarize in a compact way the semantic content of an image/video, thus requiring less bandwidth than their pixel-domain counterpart. Second, being invariant with respect to several geometric transformations (scale, rotation, illumination, viewpoint, etc...), they are particularly suited to performing analysis tasks such as object detection, recognition, tracking and classification. Finally, they can be extracted very efficiently even on low-power architectures [13]. For these reasons, feature-based visual analysis has been applied successfully to several networked scenarios, including VSNs [14] and mobile visual search [15], and constitutes the basis of the recently released MPEG-7 CDVS (Compact Descriptors for Visual Search) standard [16]. The solutions proposed for aggregating correlated image/video data in networked scenarios need to be carefully redesigned in light of such a novel feature-based paradigm. To the best of our knowledge very limited work has been done in this regard, as previous works focused only on image/video data.

In this paper we propose a prediction-based routing protocol for aggregating visual feature data in a VSN. The main contributions of this work are

as follows:

- we analyze empirically the relationship between the bitrate reduction obtained with a practical multi-view local feature encoder and several predictors. We consider different types of predictors (topology-based, image-based, feature-based and mixed ones) as well as different types of multi-view image datasets characterized by different inter-camera relationships. The purpose of the analysis is to identify the most accurate, yet compact, predictor of the achievable compression efficiency when jointly aggregating correlated streams of local features.

- we propose a new routing metric based on the identified predictor tailored to minimize the energy consumption in a VSN. The proposed routing metric is integrated in the recently standardized Routing Protocol for Low Power and Lossy Network (RPL), reusing where possible its native mechanisms.

- we evaluate the performance of the proposed methods in realistic scenarios, and we demonstrate through network simulations the benefits of feature aggregation in VSNs.

The rest of this paper is organized as follows: Section 2 discusses the related works in the area of compression and routing of correlated visual data in VSNs. In Section 3 we present an empirical analysis aimed at selecting the best predictor of the achievable multi-view feature compression (MVFC) efficiency. The resulting predictor is leveraged in Section 4 to modify the RPL protocol in order to support visual features aggregation. Section 5 provides an extensive experimental evaluation of the proposed framework and finally Section 6 concludes the paper.

2. Related Work

The broad field of data aggregation in sensor networks has always received particular attention from the research community. Many works have focused on scalar sensor networks, where data is constituted by one-dimensional measurements (e.g., temperature, humidity) and aggregation is either performed by executing simple operations (averaging a set of measurements or taking the maximum value) or relying on correlated source coding [17]. However, when it comes to Visual Sensor Networks, one should cope with the peculiar characteristics of such a technology. Being multi-dimensional in nature, visual

data is more complex than simple scalar measurements and requires ad-hoc aggregation functions. Indeed, image, video and features encoders used in such scenarios are generally complex and power-eager, and it is therefore important to consider their performance both from the aggregation efficiency and the energy consumption points of view. Several works in the recent past addressed the problem of aggregation and routing of correlated pixel-domain image data in visual sensor networks. As mentioned earlier, such works follow a two steps approach: in the first step, an estimation process is performed to predict the possible gain resulting from the joint compression of images acquired by two or more cameras with overlapping FoVs. In the second step, the prediction is leveraged to perform network-related optimizations. In [7] a spatial correlation model is proposed to describe the redundancy existing between multiple homogeneous cameras (i.e., with the same focal length). The proposed model uses geometrical information of the cameras (e.g., their locations and sensing directions) to estimate a correlation coefficient between them. The correlation coefficient is used in [18] to predict the compression efficiency of H.264 with MVC extension, and to partition a VSN into a set of coding clusters such that the global coding gain is maximized. In [9], the correlation coefficient between cameras is leveraged to set up three different network optimization problems targeting (i) the placement of multimedia processing hubs to collect and encode correlated images in a VSN, (ii) the maximization of the global compression gain and (iii) the maximization of the VSN lifetime. In [19], a correlation-aware quality-of-service routing algorithm is proposed in order to minimize energy consumption in the network subject to delay and reliability constraint. The work in [10] proposes a joint coding/routing optimization problem which maximizes the lifetime of a VSN subject to image distortion and rate constraints. Again, the key parameter in evaluating the rate-distortion function of each camera is an inter-view spatial correlation coefficient, which is assumed inversely proportional to the distance between two cameras. In such works, providing an accurate modeling of the relation between camera correlation and multi-view compression efficiency is of key importance. Therefore, several efforts have been made to improve such a modeling, either taking into account cameras heterogeneity [20], or departing from a geometric/spatial approach and taking a different approach which explicitly takes into account the visual content of the different views. In [8], the *common sensed area* (CSA) between different camera views is defined and used as a predictor for the compression efficiency of multi-view coding. Differently from previous works, the CSA

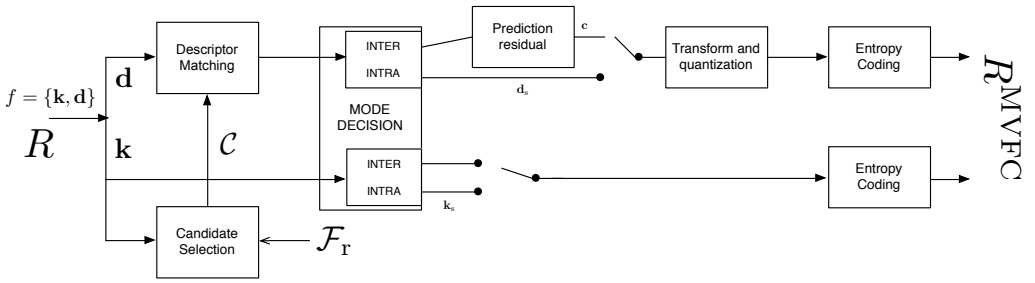


Figure 1: Multi-view features encoder architecture. Each feature f of the input feature set $\mathcal{F}$, whose original rate is R, is encoded using $\mathcal{F}_r$ as reference using $R^{\mathrm{MVFC}} < R$ bits.

is not computed based on geometric information, but is estimated starting from downsampled images which are exchanged between cameras. The main benefits in taking this approach is that it is robust to several scene-related factors (presence of moving objects, occlusions, illumination changes, etc...) that a geometric model may not capture accurately. The CSA is also leveraged in [21] to evaluate the possible benefits of a joint coding/routing scheme in multi-hop VSNs.

All the aforementioned works deal with coding and transmission of correlated *images*. To the best of our knowledge, very little work has been done targeting aggregation and routing of *features* data in networked scenarios. Some preliminary works have studied the problem of compressing local [22] or global features [23] extracted from multiple correlated views, forming the basis aggregation functions used in this work. In [24] we presented a preliminary work on aggregation and routing of local features in a VSN. In this paper we extend the study with a more detailed evaluation of the relationship between different predictors and the aggregation efficiency. In addition, we propose a practical way to implement our predictor-based approach in a VSN by integrating it into the recently standardized RPL routing protocol for low-power sensor networks.

3. MVFC Compression Efficiency Prediction

This section describes the approach taken to estimate the compression efficiency of a practical multi-view features encoder based on different predictors. First, we give a brief background on local and global features extraction and multi-view features coding. Then, we describe different predictors for the MVCF compression efficiency, including existing and novel approaches. We

compare their performance in terms of accuracy of prediction and overhead transmission cost.

3.1. Background on local and global features

There exist several different algorithms for extracting local visual features from an image, all following a two-steps approach. First, a *detector* algorithm identifies salient keypoints $\mathbf{k}$ in the image. Each keypoint is generally characterized by its location, dimension (scale) and principal orientation of the surrounding patch of pixels. Then, for each keypoint, a *descriptor* vector $\mathbf{d}$ is computed, which summarizes the photometric properties of the image area around the keypoint. A visual feature f is then composed of a keypoint and the corresponding descriptor, i.e. $f = \{\mathbf{k}, \mathbf{d}\}$, and we denote as $\mathcal{F}$ the complete set of features extracted from an image. Although the method presented in this paper could be applied to any type of features, in this work we consider features produced by the SIFT (Scale Invariant Features Transform) algorithm [25], which is widely recognized as the gold standard for a broad range of visual analysis tasks (scene analysis [26], augmented reality [27], hand gesture recognition [28], image forgery detection [29]) and has also been partially adopted by the MPEG Compact Descriptors for Visual Search (CDVS) standard [30]. Moreover, the SIFT algorithm has recently been subject of several studies aimed at optimizing its computational complexity, paving the way for its use in real-time and energy-efficient embedded machine vision systems [31, 32, 33].

The set of local features $\mathcal{F}$ can also be transformed in a global representation known as Bag of Visual Words (BoVW) according to a clustering process. Starting from a large number of representative descriptors, a clustering process similar to k-mean is executed and a vocabulary of W descriptors $\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_W$ is learned. In this case, $k = W$ and the visual words can be viewed as the centroids of the W clusters. Each descriptor in the set $\mathcal{F}$ is therefore associated with the nearest centroid. Finally, a BoVW histogram is produced: the histogram has W bins b_i, $i = 1 \ldots W$ where b_i counts the number of descriptors from $\mathcal{F}$ belonging to the cluster whose centroid is $\mathbf{w}_i$. Due to its compactness and ability to summarize efficienctly an image content, the BoVW representation is generally used in the field of content based image retrieval from very large databases [34].

3.2. Multi-view local features coding (MVFC)

Several compression algorithms have been proposed to efficiently encode a set of features $\mathcal{F}$. Mimicking the best practices in the field of video and image coding, such algorithms exploit either the redundancy of the set of features, or encode $\mathcal{F}$ using another set of features $\mathcal{F}_r$ as reference. In this work we focus on multi-view features encoders, where the sets $\mathcal{F}$ and $\mathcal{F}_r$ are extracted from two images acquired by cameras with overlapping FoVs. In particular, we take as reference design the multi-view features encoder proposed in [22], whose architecture is illustrated in Figure 1. The key idea is to use the set of features in $\mathcal{F}_r$ to predict the set of features $\mathcal{F}$. A matching step selects potential candidates for prediction. Mimicking the practices used in recent video encoders, a mode decision algorithm decides whether each feature in $\mathcal{F}$ should be encoded with respect to a local feature in $\mathcal{F}_r$ (i.e., in an inter-view coding fashion, exploiting the spatial redundancy), or intra-view coded (i.e., solely exploiting the correlation between the elements of its descriptor). In the case of inter-view coding, only the residual between the input descriptor and its best match in the base view is encoded, resorting to transform coding followed by arithmetic coding.

3.3. MVFC compression efficiency prediction

Given two sets of features $\mathcal{F}_i$ and $\mathcal{F}_j$ extracted from two cameras with overlapping fields of view, we define the MVFC coding efficiency as:

$$\eta_{i,j} = \frac{R_j - R_{i,j}^{\mathrm{MVFC}}}{R_j}. \tag{1}$$

In (1), R_j is the rate needed by camera j to encode $\mathcal{F}_j$ independently: in this case, the encoder works in intra-mode following the transform-coding scheme based on KLT transform proposed in [35]. First, the descriptors are projected in the transform domain to decorrelate their elements. Then, scalar quantization is applied to each individual descriptor element with the same quantization step. The output symbols of the quantizer are entropy coded using arithmetic coding. As for $R_{i,j}^{\mathrm{MVFC}}$ in (1), it corresponds to the rate to encode $\mathcal{F}_j$ using the MVFC encoder in inter-mode, with $\mathcal{F}_i$ as reference. In other words, $\eta_{i,j}$ is the achievable bitrate reduction on $\mathcal{F}_j$, when using $\mathcal{F}_i$ as reference with respect to intra-coding. Note that $\eta_{i,j}$ can be computed exactly only after camera i transmits $\mathcal{F}_i$ to camera j for multi-view encoding. Our goal is to estimate $\eta_{i,j}$ without explicit transmission of $\mathcal{F}_i$ to camera j,

resulting in increased network efficiency. As an example, the set of features $\mathcal{F}_i$ may be routed to the camera j for which the compression efficiency $\eta_{i,j}$ is maximized.

As it happens for multi-view coding of pixel-domain content such as images and video, we expect $\eta_{i,j}$ to be directly related to the the amount of correlation existing between the two cameras. Thus, we study the relationship of $\eta_{i,j}$ with the following predictors of inter-camera correlation:

3.3.1. Geometry-based

similarly to [7], we consider predictors based solely on the inter-camera geometry, such as the distance $d_{i,j}$ between the two camera centers or the angle $\theta_{i,j}$ between their viewing directions. The rationale behind the use of such predictors is that the closer the two camera centers and viewing directions are, the more correlated their acquired images will be and consequently the higher the resulting MVFC compression efficiency.

3.3.2. Image-based

using only the information of the inter-camera geometry may not be very accurate. Indeed, occlusions, changes in illumination and dissimilarities in the camera sensors may lead to weakly correlated images even if the two cameras are very close. It is thus reasonable to use a predictor which is able to capture the actual content of the acquired scene. Following this idea, two cameras may exchange the acquired images in order to estimate the inter-camera correlation. After such information exchange, several image-based predictor may be computed. Here we consider three different possible predictors that can be computed starting from two images:

- *Peak Signal-to-Noise Ratio ($PSNR$):* the PSNR is generally used in the field of image/video encoding to express the distortion introduced by the encoding process compared to the original content. In this work, we use the PSNR as a baseline dissimilarity metric between two images i and j and refer to it as $PSNR_{i,j}$.

- *Structural Similarity Index ($SSIM$):* the SSIM value has been proposed to overcome the limitations of the PSNR in capturing the ability of the human visual system in detecting dissimilarity between two images. In particular, $SSIM_{i,j}$ evaluates the similarities of two images i and j

based on their luminance, contrast and structure and has been demonstrated to outperform PSNR in many perceptual image and video processing applications [36].

- *Common Sensed Area (CSA):* the PSNR and SSIM were originally designed to compare impaired/compressed versions of a picture with the original one. Therefore, their power in capturing the amount of correlation between any two images is expected to be low and limited only to those cases where the fields of view of the two cameras do not differ too much. Conversely, the CSA value $\alpha_{i,j}$, originally presented in [8], captures the similarity between two images by estimating the fraction of pixels existing in the overlapping region between the view of camera i and a suitably displaced version of the view of camera j. The displacement is chosen so at to maximize the inter-view normalized bidimensional crosscorrelation function. Since the CSA is computed starting from the crosscorrelation function, rather than from a simple pixel-by-pixel comparison as done for PSNR or SSIM, it is expected to overcome the latter two in capturing the amount of similarity between two views.

The computation of the image-based predictors requires the two cameras to exchange their acquired images. To reduce the network overhead associated with such exchange, the images may be downsampled and compressed with a suitable encoder (e.g., JPEG) before transmission. In this paper, we analyze the performance of the image-based predictors considering different image resolutions as well as different JPEG compression degrees. These two factors determine the size (in bytes) of the images to exchange, therefore the associated network overhead.

3.3.3. Feature-based

The Bag of Visual Word features are widely used in the fields of content based image retrieval and augmented reality, due to their ability to summarize the content of an image. Therefore, the BoVW representations can be naturally used to understand the similarity between two images. After local feature extraction, camera i may produce a BoVW histogram and transmit it to camera j, which also computes its own histogram. In practice, the BoVW histrograms are first normalized to unit length and then quantized before transmission. Finally, the Euclidean distance between the two histograms can be computed and used as a predictor of inter-view correlation.

3.3.4. Multi-predictor approach

we also evaluate a multi-predictor approach where the MVFC compression efficiency is estimated based on the knowledge of both the geometry between the cameras and a content-based predictor.

3.4. Model fitting

We are interested in evaluating the goodness of the identified predictors in capturing the achievable MVFC compression efficiency given any two cameras. Note that the predictor should be able to identify the benefit of MVFC compression if two camera views are highly correlated, but at the same time it should reveal that there will be no benefit in compressing with the MVFC encoder two feature sets from uncorrelated/non-overlapping views. Therefore, we performed experiments relying on pairs of images obtained from cameras with both overlapping and non-overlapping fields of view. In particular, we created three image pairs datasets characterized by different features:

Linearly spaced cameras with parallel sensing directions ($d_{i,j} > 0, \theta_{i,j} = 0$). We started from the three publicly available datasets Akko&Kayo, Kendo and Balloons[1], which all contain multi-view video sequences recorded with a linear array of cameras with 5-cm spacings. From each one of the three original datasets, we chose 6 camera pairs, corresponding to a linear spacing between cameras of 5,10,15,20,25 and 30 cm respectively. For each camera pair, 50 time-synchronized frames are chosen. This gives a total number of image pairs (samples) in the newly created dataset equal to $3 \times 6 \times 50 = 900$. To populate the new dataset also with examples corresponding to camera pairs with non-overlapping fields of view, we added 900 image pairs obtained from non-synchronized frames. This is needed to test the performance of the predictor when there is no correlation between the views.

Cameras with non-parallel sensing directions ($d_{i,j} > 0, \theta_{i,j} > 0$). We started from the Columbia Object Image Library (COIL-100)[2] and Amsterdam Library of Image Objects (ALOI)[3] datasets, which contain images of objects captured at 72 different poses obtained by rotating the object by 5 degrees each time. From each dataset, 6 camera pairs are selected, corresponding

[1] http://www.fujii.nuee.nagoya-u.ac.jp/multiview-data/

[2] http://www.cs.columbia.edu/CAVE/software/softlib/coil-100.php

[3] http://aloi.science.uva.nl/

to the following angles between the camera sensing directions: {5°, 10°, 15°, 20°, 25°, 30°}. For each dataset and camera pair, 50 objects are selected, for a total of 600 image pairs. Again, to add to this new dataset examples of non-overlapping camera pairs, 600 image pairs obtained coupling images of different objects are added to this dataset.

Images from a real VSN ($d_{i,j} > 0, \theta_{i,j} > 0$). We deployed a VSN composed of 2 cameras on the roof of our university building and used them to monitor a portion of the underneath parking lot. The camera nodes are based on the design introduced in [37], using BeagleBone platforms mounting RadiumBoards cameras and wireless transceivers. The resolution of the images taken by the camera nodes is 320 × 240 pixels. The camera nodes are deployed so that the distance between their centers is 25 cm. Three different configurations are used: one in which the cameras viewing directions are parallel, one in which the angle between the viewing directions is 5 degrees and one in which the angle is 15 degrees. For each configuration 100 image pairs are selected, for a total of 300 overlapping image pairs. In addition 300 non-overlapping image pairs are added to the dataset, by selecting image couples formed by images shot in different days. The dataset is publicly available for download[4].

The tests have been performed according to the following steps: for each sample (i.e., each couple of frames (i, j) from the aforementioned datasets), we extract SIFT local features. Frame i is used as reference view, and the corresponding set of features $\mathcal{F}_i$ is used to encode the features extracted from the j-th view using the MVFC encoder. The resulting coding efficiency $\eta_{i,j}$ is stored. Simultaneously, for the same couple of frames (i, j) the following measures are computed and stored:

- the physical distance $d_{i,j}$ between the cameras or the angle between the camera sensing directions $\theta_{i,j}$, depending on the type of dataset under study. For the first two datasets, such information is given as groundtruth.

- $PSNR_{i,j}$, $SSIM_{i,j}$ and the CSA value $\alpha_{i,j}$. These measures are computed resizing the input images at different resolutions and compressing them at different JPEG quality factors. The following resolutions have

[4]http://home.deib.polimi.it/redondi/vsn/vsn.html

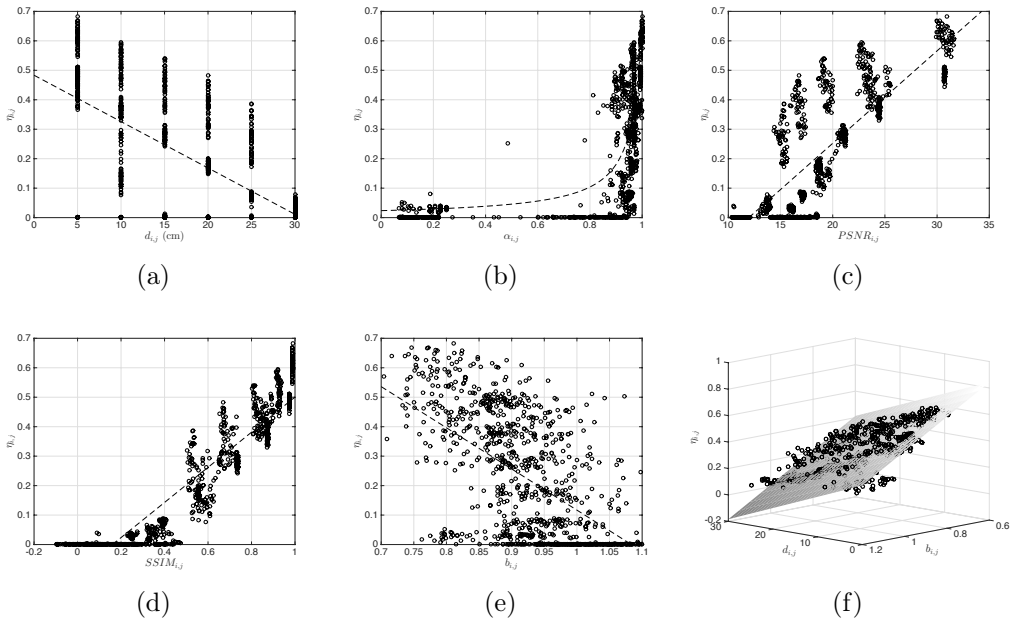


Figure 2: Different predictors of multi-view compression efficiency and fitted models. (a) Geometry-based (distance between camera centers) (b) Image-based, CSA (c) Image-based, PSNR (d) Image-based, SSIM (e) Feature-based, BoVW distance, (f) Multi-predictor approach (geometry based + feature-based)

been used: $\{(22 \times 18), (40 \times 30), (80 \times 60), (160 \times 120), (320 \times 240)\}$. For what concerns JPEG compression, ten different quality factors have been tested (from 1 to 100 with a step size equal to 10). For each combination of spatial resolution and JPEG compression factor, the average size (in bytes) of the images is stored and used to approximate the actual transmission overhead.

- the Euclidean distance between the BoVW representations of the two frames, $b_{i,j}$. We use BoVW histograms with increasing vocabulary size W in the range {128, 256, 512, 1024, 2048} visual words.

We show in Figure 2 the relationships between the MVFC compression efficiency and the different predictors for the dataset characterized by linearly spaced cameras (similar results are obtained for the datasets with non-parallel sensing directions). Figure 2(a) refers to the geometric-based predictor: the MVFC compression efficiency decreases as the distance between two cameras increases, and we fit a linear model to capture such a relationship.

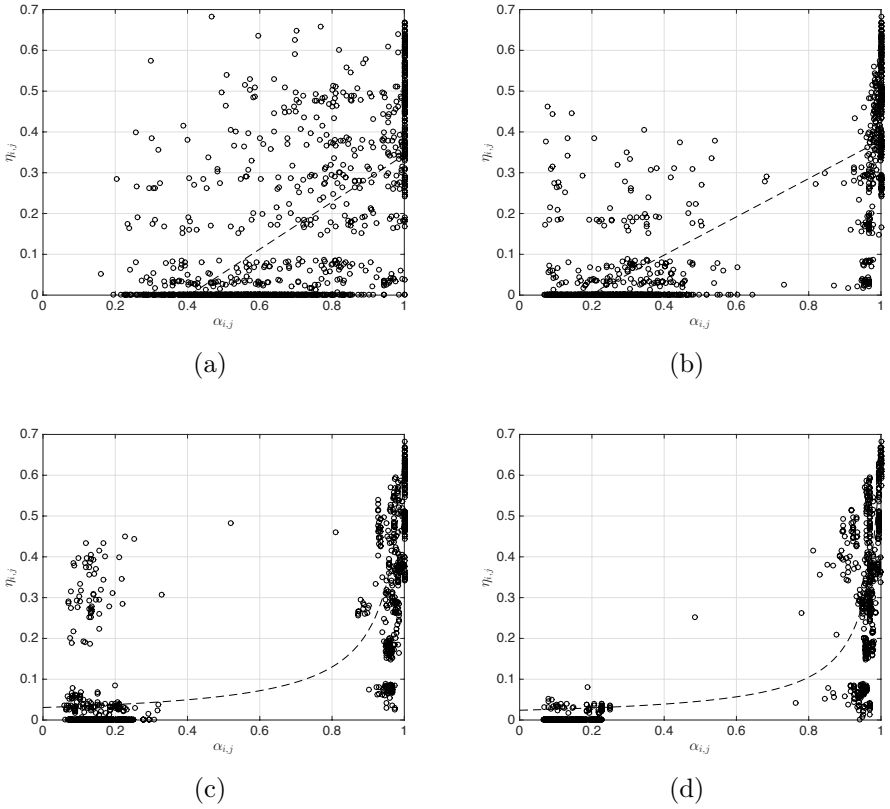


Figure 3: Performance of the CSA image-based predictor at different image resolutions and at a fixed JPEG quality factor of 10: (a) 22×18 pixels, (b) 40×30 pixels, (c) 80×60 pixels, (d) 160×120 pixels

The CSA predictor (extracted from a couple of 320×240 pixels images compressed at JPEG quality factor 10) is plotted in Figure 2(b). As one can see, the higher the CSA, the higher the compression efficiency. As proposed in [21] we use the following model to approximate the relationship between the CSA and the MVFC compression efficiency:

$$\hat{\eta}_i = \eta_{\max} \cdot \frac{\epsilon}{1 - \alpha_i + \epsilon}, \tag{2}$$

where $\eta_{\max}$ is the maximum observed compression and ϵ is a parameter to be estimated. Note that the model proposed in equation (2) does not seem to be valid for all tested pixel resolutions: Figure 3 shows how the relationship between the CSA and the compression efficiency varies when increasing the

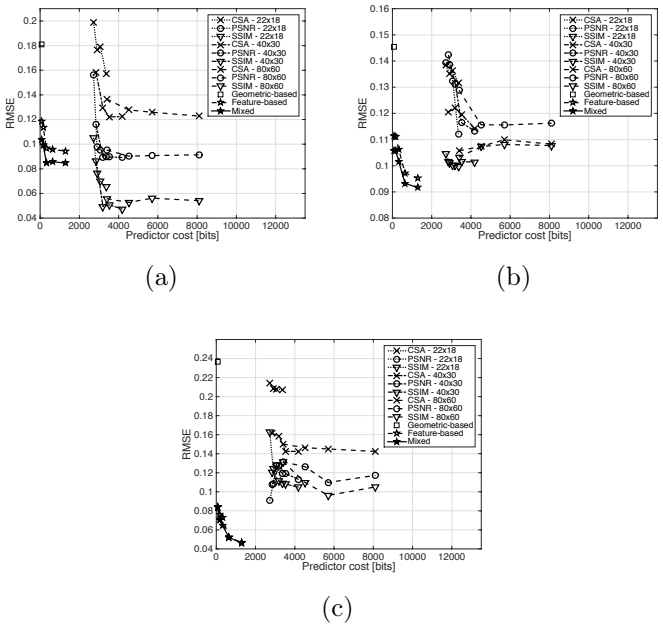


Figure 4: Performance of the different predictors in terms of their estimation accuracy and overhead transmission cost for (a) cameras with parallel sensing directions, (b) cameras with non-parallel sensing direction and (c) realistic deployment.

images resolution (at a fixed JPEG quality factor of 10). For very small resolutions (Figure 3(a) and (b)), there is no visible relationship between the CSA and η: both the RMSE and the coefficient of determination R2 calculated after fitting the model in (2) to the CSA data are lower than the ones computed with a simple linear fitting. Therefore, only for fitting CSA data, we use the model in in (2) for resolution greater than 80x60 pixels and a simple linear model for lower resolutions for the CSA case.

Figure 2(c), (d) and (e) illustrate the relationship between the MVFC compression efficiency and the PSNR value, SSIM value and BoVW distance, respectively. In all these cases, a linear model is fitted to the data points.

Finally, Figure 2(f) shows the multi-predictor approach, where the MVFC compression efficiency is estimated from both the BoVW distance $b_{i,j}$ and the physical distance $d_{i,j}$ through multilinear regression.

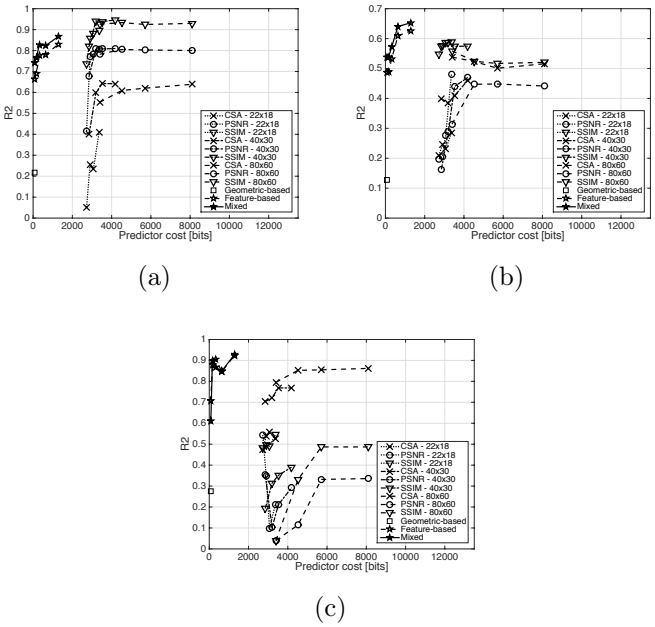


Figure 5: Performance of the different predictors in terms of their estimation accuracy and overhead transmission cost for (a) cameras with parallel sensing directions, (b) cameras with non-parallel sensing direction and (c) realistic deployment.

3.5. Predictors evaluation

To evaluate the accuracy of each predictor we resort on a traditional machine-learning based process. We divide the image datasets in a training set and a test set: the training set is used to estimate the parameters of each model, and the test set is used to compute the root mean squared error (RMSE) between the predicted and observed compression efficiency, η_i and $\hat{\eta}_i$ respectively:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\hat{\eta}_i - \eta_i)^2}, \tag{3}$$

The RMSE computation was performed relying on k-fold cross-validation with $k = 5$. For each model, we also compute the cost of transmitting the predictor between the two cameras. For the geometry-based model, one may assume that the physical topology is known to all cameras and therefore there is no need of transmitting the predictors. For the image-based models (PSNR, CSA and SSIM), we use the average size of the downsampled and

compressed images as explained in Section 3.4. For the feature-based models, the cost of the predictor depends on the BoVW size W. We assumed that each bin in the histogram is quantized uniformly using 8 bits, and the output symbols of the quantizer are lossless coded with an arithmetic encoder (whose symbols probabilities are learned from a large set of quantized BoVW histograms). Finally, for each predictor we also compute the R-squared (R2) value as a goodness-of-fit indicator.

Figures 4(a), 4(b) and 4(c) illustrate the accuracy-cost tradeoff obtained for the different predictors, on the three used datasets, while Figure 5 illustrates the R-squared value. Based on the inspection of such results, several consideration can be made:

3.5.1. Image-based methods

Different curves relative to the image-based methods at different resolutions are illustrated in Figures 4(a),(b) and (c), where each curve is obtained varying the JPEG quality factor in the range {1,20,30,60}[5]. As one can see, the performance of all image-based predictors increases with increasing image resolution (as illustrated in Figure 3). Conversely, JPEG compression hurts the performance of the image-based predictors only at very low quality factors. Among all tested image-based predictors, the SSIM is the one obtaining the lowest RMSE. At the same time, such methods requires to exchange thousands of bits of information between the cameras (from a minimum of 2700 bits for a quality factor equal to one), which can be prohibitive in certain bandwidth-constrained scenarios.

3.5.2. Geometry-based methods

If on the one hand they are the cheapest solution in terms of data transmitted between the two cameras (if the geometry is known a-priori, there is not even the need to transmit such information), on the other hand such methods have the worst performance both in terms of RMSE and goodness-of-fit. This is due to the lack in capturing the actual correlation in the content of two images, which can be null even if the two views are overlapped (as it happens in case of occlusions or difference in lightings conditions/camera lenses).

[5]we do not consider higher JPEG compression quality factors since the performance is already saturated

3.5.3. Feature-based methods

perform worse in terms of RMSE than image-based methods only for the parallel cameras dataset. For the other two more realistic datasets containing images from non-parallel cameras, they perform best. In all cases, their performance increase with the BoVW size, as expected, as a larger vocabulary can represent more accurately the similarity between two images. Also, the performance seems to saturate for vocabularies larger than 1024 words. In any case, the network overhead resulting from the use of feature-based methods is almost four times less than the one for image-based methods at the same RMSE (640 bits VS 2700 bits).

3.5.4. Multi-predictor approach

which fuses together the geometry- and feature-based methods is the one performing best both in terms of RMSE and network overhead. As one can see from Figure 4, adding geometric information to the feature-based prediction produces a benefit, although limited. This also means that the feature-based predictor is strong enough to shadow the issues caused by a wrong geometric-based prediction. For BoVW with a vocabulary size of 1024 words, the RMSE is 0.096 for the case of linearly spaced cameras and 0.103 for cameras with non parallel sensing directions and the network overhead is limited to 640 bits. This method constitutes the preferred choice and it is the one selected in the following of this work.

4. Routing protocol

In this section we describe how the proposed predictor of MVFC compression efficiency can be integrated in the Routing Protocol for Low power and Lossy networks (RPL), one of the most widespread multi-hop routing protocol for wireless sensor networks [38]. First we give a brief background on how RPL works, then we describe how it can be modified in order to support the proposed local visual features aggregation method in VSN scenarios.

4.1. Background on RPL

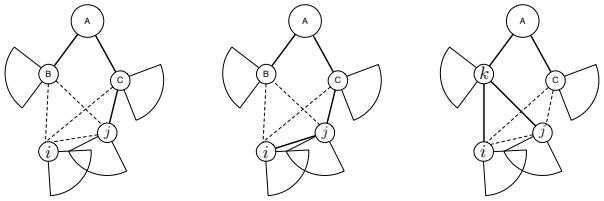
RPL was defined by the IETF ROLL (Routing Over Low-power and Lossy networks) working group and adopted as a standard since March 2012, in response to the need of creating a routing protocol for wireless sensor networks and networks of smart objects in general. Such low-power and lossy networks consist of a multitude of resource constrained nodes with limited

processing capabilities interconnected by lossy links that are usually unstable. Moreover, the traffic patterns of such networks are either Multipoint-to-Point (M2P) or Point-to-Multipoint (P2M), with sensor nodes communicating in a multi-hop fashion with a central controller node usually called sink. To address such challenges, RPL was designed as an IPv6-based distance vector routing protocol that specifies how to create a Destination Oriented Acyclic Graph (DODAG) rooted at the sink node and optimized according to a user-specific objective function (OF) working on a set of different metrics and constraints. The purpose of the OF is to give each node a Rank (a scalar representation of the depth of that node in the DODAG) and a preferred parent node (the immediate successor of the node on the path toward the sink). This is done through the exchange of particular signalling messages called DODAG Information Object (DIO) messages, which carry information that allows a node to join a DODAG, compute its rank and select its preferred parent node. For the common case of multipoint-to-point traffic (where data traffic flows from sensor nodes to the sink), the macro-steps involved in the creation of a DODAG are the following:

1. The sink node starts sending DIO messages advertising the DODAG. In its DIOs, the root node advertizes its own rank as the minimum one (equal to 0).
2. Nodes receive DIOs from their in-range neighbours (referred to as neighbouring set $\mathcal{N}$). To avoid loops in the DODAG, nodes process the DIOs of neighbours characterized by lower ranks (the so-called candidate parents set $\mathcal{P}$). After that, nodes compute their ranks according to the selected OFs and select their preferred parent among the candidate parents set. Then, they update the DIOs with their own rank and routing metric and forward the DIO to other sensors.

Besides the traditional shortest path distance vector routing, where nodes join the DODAG so as to reach the sink node in the minimum number of hops, one of the most used objective function/metric combination in RPL is the minimization of the Expected Transmission Count (ETX), a link metric that represents the average number of times a packet should be transmitted in order to successfully reach its destination. In order to find a path from a node to the root of the DODAG with the least number of transmissions, a non-root node i computes the ETX path metric for a path to the root through each candidate parent j by adding these two components:

1. The ETX $\epsilon_{i,j}$ on the link from i to j. This is generally estimated as



(a) (b) (c)

Figure 6: A VSN in which nodes i and j shares part of their FoVs. Solid lines represents already computed paths, dashed lines represents communication links. (a) Node j is already associated to the network and has a path to the sink (b) Node i choses j as selected parent and its features will be aggregated there. (c) Node i selects k as a selected parent. Node j also switch to k in order to aggregate its features with node i's ones.

$\epsilon_{i,j} = \frac{1}{P_f P_r}$, where P_f is the probability of successful packet delivery and P_r is the the probability of successful acknowledgment reception.

2. The cumulative ETX from j to the root node, $\epsilon_{j,\mathrm{r}}$, which is advertised by node j in its DIO messages.

Then, node i selects the candidate parent for which the cumulative ETX path metric is minimized.

The ETX itself can be also converted from a link quality metric to an energy metric, capturing the total energy cost that a packet transmission causes on the entire network on its path from the source node to the sink. In particular, assuming that E^{tx} is the per-packet transmission energy consumption of a node, the product $E^{\mathrm{tx}} \cdot \epsilon_{i,j}$ captures the average transmission energy spent by node i to transmit one packet to node j. Similarly, $E^{\mathrm{tx}} \cdot \epsilon_{j,r}$ is the average transmission energy spent to transmit one packet from node j to the sink. Consequently, by adopting such a conversion, it is possible to select paths in a DODAG so that the overall amount of energy consumed for the transmission process is minimized.

4.2. RPL for features aggregation

In order to use RPL as a routing protocol for supporting features aggregation in VSNs, we propose a routing metric able to identify paths where such aggregation can take place, as well as the mechanisms needed for the computation of the metric itself.

We observe that performing features aggregation may be beneficial to two aspects of a VSN: *bandwidth* and *energy*. Bandwidth can be saved if

camera nodes route their visual features to nodes with correlated views, so that multi-view compression may be applied. In turn, the reduction in the amount of data to forward toward the sink node directly reflects on the consumed energy for data transmission/reception of (battery-operated) relay nodes, thus extending the network lifetime. However, features compression does not come for free: running the MVFC encoder summarized in Section 3.2 requires a non-negligible energy cost E_{c} which can impact negatively on energy consumption. Indeed, a camera node attempting to aggregate (compress) its features with the ones from another node without any hint on the amount of correlation between the two sets of features may end up paying an extra energy cost for compression without any benefit in terms of saved bandwidth. Our proposed predictor of MVFC compression efficiency is intended to solve this dilemma and avoid such unpleasant scenarios.

Figure 6(a) shows a VSN with two nodes with overlapping FoVs. Let i be a camera node that needs to join an existing DODAG, and let $\hat{\eta}_{i,j}$ be the compression efficiency estimated by i after reception of the predictor from all nodes in its neighbouring set $\mathcal{N}_i$ (that is nodes j, B and C in Figure 6(a)). Camera node i needs to transmit R_i bits of information, containing the visual features data extracted from an image, which corresponds to N_i radio packets. Should i route its features to j, the estimated number of bits for i's features after feature compression is $(1 - \hat{\eta}_{i,j}) \cdot R_i$, corresponding to a number of radio packets $M_i \leq N_i$.

It is useful to define two variables for capturing the average energy consumption spent by the entire network in the process of collecting features from different camera nodes, namely the *energy for independent transmission*:

$$E_i = E^{\mathrm{tx}} \epsilon_{i,r} N_i, \tag{4}$$

where E_i captures the energy spent for features transmission from node i to the sink using a preferred path without aggregation and the *energy for transmission with aggregation*:

$$E^k_{i,j} = E^{\mathrm{tx}} \epsilon_{i,k} N_i + E^{\mathrm{tx}} \epsilon_{j,k} N_j + E^{\mathrm{tx}} \epsilon_{k,r} (M_i + N_j) + E_{\mathrm{c}} \tag{5}$$

where $E^k_{i,j}$ denotes the energy spent when node i aggregates its features with the ones from node j at node k. Note that $E^k_{i,j}$ captures the costs associated to both i's and j's feature streams, as well as the additional cost for performing MVFC coding at k, E_{c}. Note also that k in this case must be either equal to

j or a common candidate parent of both i and j. Figure 6(b) and (c) provide a graphical illustration of these scenarios, respectively.

We propose a routing metric that selects paths so that the energy saved through features aggregation is maximized. In order to do this, a node i computes the values:

$$G_j^k = E_i + E_j - E_{i,j}^k \tag{6}$$

for all neighbours j and considering all possible common candidate parents k. The measure G_j^k captures the total amount of energy saved in the network if i and j aggregate their features at k. Note that, in order to compute G_j^k and agree on a common parent k, node i and j both have to:

- Estimate the MVFC coding efficiency $\hat{\eta}_{i,j}$ exchanging the 640-bit predictor and using the model illustrated in Figure 2(e) (or in Figure 2(f), if geometric information is as well available).

- Compute the total energy cost for independent transmission, $E_i + E_j$, using equation (4). This requires that i and j exchange both the cumulative ETX on their selected path to the sink, $\epsilon_{i,r}$ and $\epsilon_{j,r}$ and the number of features packets to transmit N_i and N_j.

- Compute the potential energy saved through features aggregation G_j^k for all common candidate parents k using equation (5). This requires that i and j exchange their set of candidate parents $\mathcal{P}_i$ and $\mathcal{P}_j$. The set of common parents is then $\mathcal{C}_{i,j} = \mathcal{P}_i \cap \mathcal{P}_j$.

- Select a node j to aggregate features with and the selected parent k at which such aggregation must be performed:

$$\max_{j \in \mathcal{N}_i, k \in \mathcal{C}_{i,j}} G_j^k \tag{7}$$

- Finally, node i sends its features to k and mark its packets to be aggregated with the ones from j.

4.3. DIO message structure

A natural choice for exchanging the information required for computing the values in (4)-(6) among cameras using RPL mechanisms is to embed them in the DAG metric container option present in DIO messages, which can be used to report metrics along the DODAG as chosen by the implementer. In details, each node i should add the following information to its DIO messages:

- The 640-bit predictor of MVFC compression efficiency (80 bytes).
- The current selected parent k and the cumulative ETX for reaching the sink through k, $\epsilon_{i,r}$. According to RPL specifications, the ETX value for a link is encoded using 16 bits in unsigned integer format. The address of the selected parent is encoded using 16-bit short addresses, which are unique within the network according to the IEEE 802.15.4 standard. Transferring this information requires 4 bytes.
- The set of alternative possible candidate parents $\mathcal{P}_i$ and the ETX on the links to them $\epsilon_{i,p}$, $p \in \mathcal{P}_i$, for a total of 4 bytes per candidate parents. Assuming that DIO messages are encapsulated in IEEE 802.15.4 frames with header compression, which allows for a payload size of 97 bytes, the number of alternative possible candidate parents each node may disseminate in a single frame is limited to $\lfloor (97 - 80 - 4)/4 \rfloor = 3$. Therefore, we propose to sort the set of candidate parents in increasing order of their associated ETX before DIO dissemination and to transmit the best three candidate parents.

4.4. Extension to other types of data

With the proposed changes, the RPL protocol can be used to support aggregation not only of local visual features, but also of any type of data. Both scalar (e.g., temperature, humidity) data or complex measurements (e.g. pixel-domain images, videos) could in fact be aggregated on their path to the sink node, following the same principles used in this work. Note that, for adapting the proposed scheme to a specific type of data, one needs to define (i) a proper aggregation function, (ii) the energy cost needed to execute such aggregation function and (iii) a predictor able to estimate the benefit of aggregation for each pair of nodes. As an example, for the case of transmission of pixel-domain images or videos, the MVC extension of the H.264/MPEG ACV encoder could serve as aggregation function (with its corresponding energy cost) and the CSA between the two views could be used as a predictor, as proposed in [21].

5. Experimental Evaluation

To evaluate the performance of the proposed framework, we have carried out extensive experimental simulations. In particular, we are interested in as-

sessing how beneficial features aggregation is in realistic scenarios, compared to the following cases:

1. camera nodes compress and transmit their features to the sink in an independent way, using only intra-mode encoding and without resorting to aggregation. In this case each camera computes the best path to the sink by minimizing the cumulative ETX.
2. camera nodes perform opportunistic aggregation. In details, cameras still computes the best path to the sink by minimizing the cumulative ETX. Intermediate cameras on the path to the sink always run the MVFC encoder trying to aggregate their own features with the one received from their children.

For the task at hand, we simulated several VSN instances characterized by different numbers of camera nodes. Camera nodes are added to the simulation in couples, reflecting the actual configurations of cameras in the datasets used in Section 3. Therefore, half of the camera couples are added so that their fields of view are non-overlapping. The other half of camera couples is deployed so that the distance between the camera centers is randomly selected in the range 5-30 cm and the viewing direction is chosen so that half of the camera couples (25% of the initial couples) have parallel viewing directions and the other half have viewing directions chosen uniformly in the range 5°.-°.. The camera couples are then deployed randomly in a 100m $\times$ 100m simulation area, as illustrated in Figure 7. Finally, a single sink node is deployed at the center of the simulation area. Each node in the sensor network is modeled based on a real-life implementation of a camera node: in particular, we rely on a Linux-operated BeagleBone Black platform, which is coupled with a IEEE 802.15.4-compliant Memsic TelosB dongle and with an ad-hoc camera board. The platform runs an open-source framework for VSNs capable of performing several processing tasks, including features extraction and multi-view features encoding [37].

According to the chosen platform, in our simulation the MAC and PHY layers are compliant with the IEEE 802.15.4 specifications and the network layer runs IPv6 RPL. At the application layer, a simple UDP agent is used to periodically transmit features to the sink. The channel model used in the simulation is the Unit Disk Graph Medium (UDGM) distance loss model, according to which a link is established between two cameras if their distance is less than a pre-defined communication range, which depends on the output transmission power and is set equal to 30 meters in our tests.

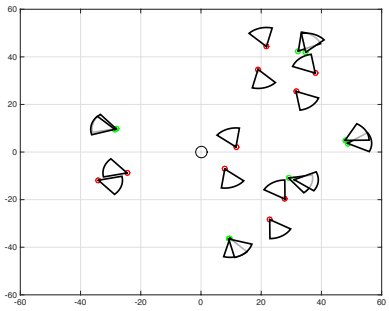


Figure 7: A network instance with 20 cameras used in our simulations. The big black circle at the center of the area represents the sink node. Half of the deployed camera couples have overlapping fields of view (green dots) and half have non-overlapping fields of view (red dots).

For each couple of cameras with overlapping fields of view in one network instance, we randomly select a couple of images from our datasets: the selection process is performed so that the distance between camera centers and the angle between their viewing direction in the simulation is approximately the same as the ones characterizing the selected images. For camera couples with non-overlapping fields of view, uncorrelated images are selected (e.g., images of different objects). Finally, for each couple of cameras and selected images, we generate the actual aggregation parameter $\eta_{i,j}$ by running the MVFC encoder and we compute the estimated aggregation parameter $\hat{\eta}_{i,j}$ using the multi-predictor model.

For each network instance, the simulation keeps track of two performance metrics: *total used bandwidth* and *total consumed energy*. The former is measured counting the aggregated number of feature data packets transmitted in the network, including both packets generated by cameras and packets forwarded by relay nodes. The latter metric is measured by relying on a realistic energy model which captures all the operations performed on-board cameras. In particular, we rely on the following model for tracking the energy consumption at camera i:

$$E_i^{\text{tot}} = E^{\text{tx}} M_i + E^{\text{rx}} \sum_{j \in \mathcal{C}_i} N_j + n_i E_{\text{c}}, \tag{8}$$

where M_i is the total number of packets transmitted by i, including both i's data packets and packets aggregated/forwarded from its children (in the

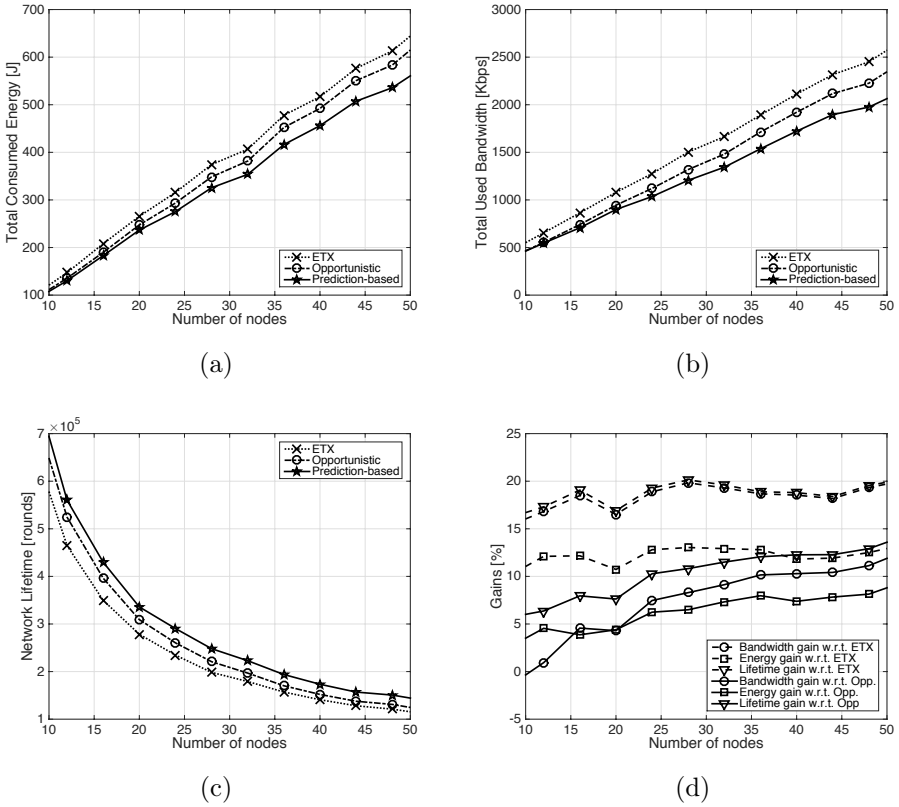


Figure 8: (a) Total energy consumed, (b) total used bandwidth per round and (c) achievable network lifetime at different number of camera nodes in the network. Results are averaged over 50 experiments. (d) Energy, bandwidth and lifetime gains achievable by the proposed approach with respect to ETX or opportunistic aggregation.

set $\mathcal{C}_i$), N_j is the number of packets received by i from its j-th children and n_i is the number of children that aggregate their features at i (the cost of multi-view encoding should be paid for each children).

The values used for the transmission and reception energy consumptions E^{tx}, E^{rx} and for the multi-view coding energy consumption E_{c} have been measured indirectly by keeping track of the time spent by the BeagleBone-based visual sensor node in each operative mode and multiplying this time by the platform power consumption. The obtained values are summarized in Table 1.

The simulations are performed following these steps:

Table 1: Measurements from a VSN testbed

Name	Symbol	Value
Per-packet transmission cost	E^{tx}	5.48×10^{-2} J/packet
Per-packet reception cost	E^{rx}	5.62×10^{-2} J/packet
Energy cost for multi-view coding	E^{c}	2.14×10^{-1} J

- The camera network is deployed randomly in the area, and each node but the sink is given an initial energy budget E_{b} equal to 32.4×10^3 J, corresponding to the energy available to a BeagleBone-based visual sensor node when powered with four standard AA batteries (1.5V, 1500 mAh).

- Cameras are periodically fed with new images. Upon image generation and features extraction, each camera runs RPL to find routing paths to the sink. In the first test, camera nodes use only the ETX as routing metric and do not perform any type of aggregation. In a second test, camera nodes still use only the ETX as a routing metric, but they also perform opportunistic aggregation, running the MVFC encoder on all incoming features set. In the third test, camera nodes run the proposed version of RPL using the predictor-based approach. Note that in the first two tests the routing tree does not change, while in the third test the routing tree may change from round to round due to the predictor-based approach which routes data so that the benefit of aggregation is maximized.

- At each round (generation of images), features are transmitted to the sink node along the computed routing tree. It is hence possible to compute the total used bandwidth and the total consumed energy. Note that, for the prediction-based approach, the routing tree is computed using the *estimated* aggregation parameter $\hat{\eta}_{i,j}$, while the total used bandwidth and the total consumed energy are computed using the *actual* aggregation parameter $\eta_{i,j}$. The simulation runs until the first node in the network depletes its energy budget (determining the network lifetime). The experiment is repeated 50 times varying the position of the cameras, and results are averaged.

Figures 8(a) and (b) show the average total consumed energy and total

used bandwidth per round for different number of camera nodes in the network. As one can see, for all methods tested, the energy consumption and utilized bandwidth increase as the number of nodes in the network increases. The ETX metric is the one consuming most energy and bandwidth, followed by opportunistic aggregation, while the proposed predictor-based approach is the one performing best. Figure 8(c) shows the network lifetime achievable by the three different methods: being the most energy-eager, the ETX allows for the smallest lifetime. Conversely, the proposed approach based on prediction obtains the best results. Finally, Figure 8(d) shows the bandwidth, energy and lifetime gains achievable by the proposed approach with respect to using the ETX or opportunistic aggregation. As one can see, compared to the ETX, the predictor-based approach is able to save up to 20% bandwidth and lifetime, by carefully selecting paths to the sink in order to maximize the benefit of aggregation. The total energy savings compared to the ETX are smaller, in the order of 12%: this is due to the additional costs spent by camera nodes for (i) disseminating the predictors in their DIOs and (ii) running the multi-view features encoder. Overall, the gains obtained by the proposed approach over the ETX metric seems independent from the network size (in terms of number of nodes). Differently, the gain achievable by the proposed approach over opportunistic aggregation increases as the number of node increases. This can be explained considering that in opportunistic aggregation a node always runs the MVFC encoder on the features received from its children, in a blind way: this is inefficient from an energy point of view and such inefficiency increases as the number of nodes in the network increases. Finally, the importance of selecting an accurate predictor is illustrated in Figure 9, where the energy, bandwidth and lifetime gains of the proposed approach over the ETX and opportunistic aggregation are reported at different predictor RMSE, selected in the range of the ones found in our evaluation described in Section 3. The experiment is performed over 50 instances of VSNs composed of 40 camera nodes and results are averaged. As one can see, as the RMSE increases, the achievable gains decrease rapidly. This is caused by the routing algorithm which selects non-optimal paths due to the inaccuracy of the predictor.

6. Conclusions

In this paper, we propose a methodology for jointly routing and compressing streams of local features in visual sensor networks. The method-

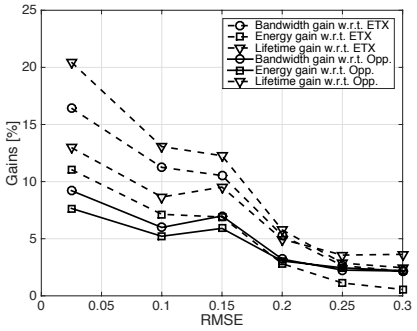


Figure 9: Achievable gains at different predictor RMSE

ology exploits a predictor to identify cameras with similar visual content, which is carefully identified among different geometric-based, image-based and feature-based predictors. The selected predictor allows to estimate with good accuracy the compression efficiency, with at an overhead transmission cost of less than 700 bits. Therefore it is very well suited to be used in applications for bandwidth and energy constrained networks. Secondly, we integrate the selected predictor in the working operation of the RPL protocol for low power and lossy networks, obtaining a protocol able to support multi-view features aggregation. We carefully modify the protocol routing metrics in order to take into account the peculiarity of such a scenario and we demonstrate through experiments the benefits of the proposed approach.

[1] K. Džubáková, P. Molnar, K. Schindler, M. Trizna, Monitoring of riparian vegetation response to flood disturbances using terrestrial photography, Hydrology and Earth System Sciences 19 (1) (2015) 195–208.

[2] A. Miguel, S. Beery, E. Flores, L. Klemesrud, R. Bayrakcismith, Finding areas of motion in camera trap images, in: Image Processing (ICIP), 2016 IEEE International Conference on, IEEE, 2016, pp. 1334–1338.

[3] Y. Yim, H. Cho, S.-H. Kim, E. Lee, M. Gerla, Vehicle location service scheme based on road map in vehicular sensor networks, Computer Networks.

[4] E.-K. Lee, J.-H. Lim, M. Gerla, Polycast: A new paradigm for information-centric data delivery in heterogeneous mobile fog networks, International Journal of Distributed Sensor Networks 13 (9).

[5] A. Vetro, T. Wiegand, G. J. Sullivan, Overview of the stereo and multiview video coding extensions of the h.264/mpeg-4 avc standard., Proceedings of the IEEE 99 (4) (2011) 626–642.
URL http://dblp.uni-trier.de/db/journals/pieee/pieee99.html#VetroWS11

[6] B. Girod, A. Aaron, S. Rane, D. Rebollo-Monedero, Distributed video coding, Proceedings of the IEEE 93 (1) (2005) 71–83. doi:10.1109/JPROC.2004.839619.

[7] R. Dai, I. Akyildiz, A spatial correlation model for visual information in wireless multimedia sensor networks, Multimedia, IEEE Trans. on 11 (6) (2009) 1148–1159. doi:10.1109/TMM.2009.2026100.

[8] S. Colonnese, F. Cuomo, T. Melodia, Leveraging multiview video coding in clustered multimedia sensor networks, in: Global Communications Conf. (GLOBECOM), 2012 IEEE, 2012, pp. 475–480. doi:10.1109/GLOCOM.2012.6503158.

[9] P. Wang, R. Dai, I. Akyildiz, Visual correlation-based image gathering for wireless multimedia sensor networks, in: INFOCOM, 2011 Proceedings IEEE, 2011, pp. 2489–2497. doi:10.1109/INFCOM.2011.5935072.

[10] C. Li, J. Zou, H. Xiong, C. W. Chen, Joint coding/routing optimization for distributed video sources in wireless visual sensor networks, Circuits and Systems for Video Technology, IEEE Trans. on 21 (2) (2011) 141–155. doi:10.1109/TCSVT.2011.2105596.

[11] R. Dai, P. Wang, I. Akyildiz, Correlation-aware qos routing for wireless video sensor networks, in: Global Telecommunications Conf. (GLOBECOM 2010), 2010 IEEE, 2010, pp. 1–5. doi:10.1109/GLOCOM.2010.5684202.

[12] A. Redondi, L. Baroffio, L. Bianchi, M. Cesana, M. Tagliasacchi, Compress-then-analyze vs analyze-then-compress: what is best in visual sensor networks?, IEEE Transactions on Mobile Computing PP (99) (2016) 1–1. doi:10.1109/TMC.2016.2519340.

[13] L. Baroffio, A. Canclini, M. Cesana, A. Redondi, M. Tagliasacchi, Briskola: Brisk optimized for low-power arm architectures, in: 2014

IEEE International Conference on Image Processing (ICIP), IEEE, 2014, pp. 5691–5695.

[14] A. Redondi, L. Baroffio, M. Cesana, M. Tagliasacchi, Compress-then-analyze vs. analyze-then-compress: Two paradigms for image analysis in visual sensor networks, in: Multimedia Signal Processing (MMSP), 2013 IEEE 15th Intl. Workshop on, 2013, pp. 278–282. doi:10.1109/MMSP.2013.6659301.

[15] B. Girod, V. Chandrasekhar, D. Chen, N.-M. Cheung, R. Grzeszczuk, Y. Reznik, G. Takacs, S. Tsai, R. Vedantham, Mobile visual search, Signal Processing Magazine, IEEE 28 (4) (2011) 61–76. doi:10.1109/MSP.2011.940881.

[16] L.-Y. Duan, F. Gao, J. Chen, J. Lin, T. Huang, Compact descriptors for mobile visual search and mpeg cdvs standardization, in: Circuits and Systems (ISCAS), 2013 IEEE Intl. Symposium on, 2013, pp. 885–888. doi:10.1109/ISCAS.2013.6571989.

[17] H. Luo, G. J. Pottie, Designing routes for source coding with explicit side information in sensor networks, IEEE/ACM Transactions on Networking 15 (6) (2007) 1401–1413. doi:10.1109/TNET.2007.900703.

[18] P. Wang, R. Dai, I. Akyildiz, Collaborative data compression using clustered source coding for wireless multimedia sensor networks, in: INFOCOM, 2010 Proceedings IEEE, 2010, pp. 1–9. doi:10.1109/INFCOM.2010.5462034.

[19] R. Dai, P. Wang, I. Akyildiz, Correlation-aware qos routing with differential coding for wireless video sensor networks, Multimedia, IEEE Trans. on 14 (5) (2012) 1469–1479. doi:10.1109/TMM.2012.2194992.

[20] M. Y. Mowafi, F. H. Awad, W. A. Aljoby, A novel approach for extracting spatial correlation of visual information in heterogeneous wireless multimedia sensor networks, Computer Networks 71 (2014) 31 – 47. doi:http://dx.doi.org/10.1016/j.comnet.2014.06.010.
URL http://www.sciencedirect.com/science/article/pii/S1389128614002412

[21] S. Colonnese, F. Cuomo, T. Melodia, An empirical model of multiview video coding efficiency for wireless multimedia sensor networks, Multimedia, IEEE Trans. on 15 (8) (2013) 1800–1814. doi:10.1109/TMM.2013.2271475.

[22] L. Bondi, L. Baroffio, M. Cesana, A. Redondi, Tagliasacchi, Multi-view coding of local features in visual sensor networks, in: IEEE Intl. Conf. on Multimedia and Expo (ICME) - Workshop on Distributed and Cooperative Visual Recognition and Analysis (DCVRA), 2015.

[23] N. Naikal, A. Y. Yang, S. S. Sastry, Towards an efficient distributed object recognition system in wireless smart camera networks, in: 13th Conf. on Information Fusion, FUSION 2010, Edinburgh, UK, July 26-29, 2010, IEEE, 2010, pp. 1–8.
URL http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=5711893

[24] A. E. Redondi, L. Baroffio, M. Cesana, M. Tagliasacchi, Multi-view coding and routing of local features in visual sensor networks, in: IEEE INFOCOM 2016-The 35th Annual IEEE International Conference on Computer Communications, IEEE, 2016, pp. 1–9.

[25] D. G. Lowe, Distinctive image features from scale-invariant keypoints, Intl. Journal of Computer Vision 60 (2) (2004) 91–110. doi:10.1023/B:VISI.0000029664.99615.94.
URL http://dx.doi.org/10.1023/B:VISI.0000029664.99615.94

[26] A. Bosch, A. Zisserman, X. MuÃśoz, Scene classification using a hybrid generative/discriminative approach, IEEE Transactions on Pattern Analysis and Machine Intelligence 30 (4) (2008) 712–727. doi:10.1109/TPAMI.2007.70716.

[27] D. Wagner, G. Reitmayr, A. Mulloni, T. Drummond, D. Schmalstieg, Real-time detection and tracking for augmented reality on mobile phones, IEEE Transactions on Visualization and Computer Graphics 16 (3) (2010) 355–368. doi:10.1109/TVCG.2009.99.

[28] N. H. Dardas, N. D. Georganas, Real-time hand gesture detection and recognition using bag-of-features and support vector machine techniques, IEEE Transactions on Instrumentation and Measurement 60 (11) (2011) 3592–3607. doi:10.1109/TIM.2011.2161140.

[29] I. Amerini, L. Ballan, R. Caldelli, A. D. Bimbo, G. Serra, A sift-based forensic method for copy and move attack detection and transformation recovery, IEEE Transactions on Information Forensics and Security 6 (3) (2011) 1099–1110. doi:10.1109/TIFS.2011.2129512.

[30] L.-Y. Duan, F. Gao, J. Chen, J. Lin, T. Huang, Compact descriptors for mobile visual search and mpeg cdvs standardization, in: 2013 IEEE International Symposium on Circuits and Systems (ISCAS2013), IEEE, 2013, pp. 885–888.

[31] F.-C. Huang, S.-Y. Huang, J.-W. Ker, Y.-C. Chen, High-performance sift hardware accelerator for real-time image feature extraction, IEEE Transactions on Circuits and Systems for Video Technology 22 (3) (2012) 340–351.

[32] J. Jiang, X. Li, G. Zhang, Sift hardware implementation for real-time image feature extraction, IEEE Transactions on Circuits and Systems for Video Technology 24 (7) (2014) 1209–1220.

[33] Z. Liu, F. Qiao, Q. Wei, X. Yang, Y. Li, H. Yang, An ultra-fast and low-power design of analog circuit network for dog pyramid construction of sift algorithm, in: 2016 17th International Symposium on Quality Electronic Design (ISQED), 2016, pp. 392–397. doi:10.1109/ISQED.2016.7479233.

[34] J. Sivic, A. Zisserman, Video Google: A text retrieval approach to object matching in videos, in: Proceedings of the Intl. Conf. on Computer Vision, Vol. 2, 2003, pp. 1470–1477.
URL http://www.robots.ox.ac.uk/~vgg

[35] L. Baroffio, M. Cesana, A. Redondi, M. Tagliasacchi, S. Tubaro, Coding visual features extracted from video sequences, IEEE Transactions on Image Processing 23 (5) (2014) 2262–2276.

[36] Z. Wang, A. C. Bovik, H. R. Sheikh, E. P. Simoncelli, Image quality assessment: from error visibility to structural similarity, IEEE Transactions on Image Processing 13 (4) (2004) 600–612. doi:10.1109/TIP.2003.819861.

[37] L. Bondi, L. Baroffio, M. Cesana, A. Redondi, M. Tagliasacchi, Ez-vsn: An open-source and flexible framework for visual sensor networks, IEEE

Internet of Things Journal 3 (5) (2016) 767–778. doi:10.1109/JIOT.2015.2504622.

[38] T. Winter, Rpl: Ipv6 routing protocol for low-power and lossy networks, IETF Request for Comments: 6550.