Characterization of SURF and BRISK Interest Point Distribution for Distributed Feature Extraction in Visual Sensor Networks

György Dán, Muhammad Altamash Khan, Viktoria Fodor
ACCESS Linnaeus Center, School of Electrical Engineering
KTH Royal Institute of Technology
Stockholm, Sweden
{gyuri,khan6,vfodor}@ee.kth.se

***Abstract*—We study the statistical characteristics of SURF and BRISK interest points and descriptors, with the aim of supporting the design of distributed processing across sensor nodes in a resource constrained visual sensor network.**

Our results show high variability in the density, in the spatial distribution, and in the octave layer distribution of the interest points. The high variability implies that balancing the processing load among the sensor nodes is a very challenging task, and obtaining a-priori information is essential, e.g., through prediction. Our results show that if a-priori information is available about the images, then Top-M interest point selection, limited, octave-based processing at the camera node, together with area-based interest point detection and extraction at the processing nodes can balance the processing load and limit the transmission cost in the network. Complete interest point detection at the camera node with optimized descriptor extraction delegation to the processing nodes in turn can further decrease the transmission load and allow a better balance of the processing load among the network nodes.

***Index Terms*—SURF; BRISK; visual sensor network; interest point distribution; distributed feature extraction**

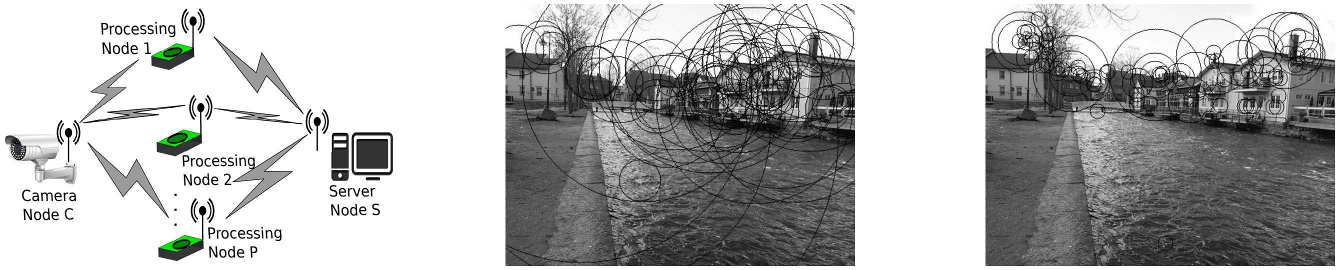
I. Introduction

In visual sensor networks (VSNs), images or video sequences captured by low cost cameras need to be processed and transmitted to a central node or image database to perform visual analysis tasks, like image retrieval or video data mining [1]. Visual analysis is often based on the detection and extraction of visual features that can characterize the image. The detected set of features should enable high performance visual analysis, for example, in the case of image retrieval high precision and recall. Moreover, the features should be scale and rotation invariant, so that objects captured from different distances and angles can be identified [2], [3], [4], [5].

In general, feature extraction techniques consider the pixel data of an image, detect a set of interest points and extract the related feature descriptors. The feature descriptors are then used to, e.g., retrieve images or identify objects in a database. For visual analysis in a VSN environment one of two approaches is usually followed, compress-then-analyze or analyze-then-compress. Under the compress-then-analyze approach the image is compressed in pixel domain and is transmitted to the central node, which performs the feature extraction. The key challenge is to minimize the amount of pixel data to be transmitted. As lossy coding in the pixel domain affects the visual analysis at the central node, recent works propose image coding schemes optimized for feature extraction [6], [7].

Under the analyze-then-compress approach the camera node performs the feature extraction, and the compressed descriptors are transmitted to the central node. The main challenge is to minimize the computational load at the camera node, and therefore many of the emerging feature extraction schemes aim at decreasing the computational complexity [8], [9]. If the transmission bandwidth of the VSN is very limited, the objective can even be to limit the amount of data to be transmitted by compressing the descriptors or by limiting their number. Descriptor compression techniques are suggested and evaluated in [10], [11]. In [12] a progressive transmission scheme is proposed, which terminates the transmission of new descriptors as soon as the image is retrieved. In [13] the number of considered interest points and the quantization level of the descriptors are jointly optimized to maximize the accuracy of the recognition, subject to energy and bandwidth constraints. In the case of visual analysis of video sequences the computational load and the transmission requirements can be decreased by leveraging the temporal correlation among the frames. In [14] candidate interest points are selected based on motion prediction. As only the areas around these candidates are processed, the computational load is decreased. In [15] intra- and inter-frame coding of descriptors is proposed to decrease the transmission requirements.

Both of these schemes, the compress-then-analyze and the analyze-then-compress, use the nodes of the VSN exclusively for forwarding data to the central node. Nevertheless, VSN nodes that do not have a camera could in principle be leveraged for in-network data processing. Sensor nodes without a camera are arguably cheaper, they could thus be used to increase the computational power available in the sensor network at a relatively low cost. Furthermore, processing in the VSN nodes could be used to decrease the computational load and the corresponding energy consumption of the camera nodes, and can thus help to extend their lifetimes. While camera



(a) Considered VSN scenario.

(b) Interest areas for SURF interest points

(c) Interest areas for BRISK interest points

Fig. 1: Considered VSN scenario and example for SURF and BRISK interest point detection.

nodes need to be parametrized and calibrated in situ in order to be able to capture images for complex 3D visual analysis tasks, such as tracking, processing nodes can more easily be deployed. It would thus be favorable for a VSN to drain the batteries of the processing nodes and to replace those more frequently, instead of replacing the camera nodes. The feasibility of this approach in a multihop VSN that consists of battery driven BeagleBone Black single board computers was recently demonstrated for object recognition and tracking using BRISK in [16].

In this paper we consider that the nodes of the VSN have computational capabilities, and we evaluate the possible gains of utilizing the VSN nodes not only for transmission, but also for in-network interest point detection and feature extraction. We take into account constraints pertinent to VSNs, in particular, the limited transmission rate, computational power and energy resources of the nodes, which call for the optimization of the communication as well as the processing in the VSN. We consider two popular but conceptually different feature extraction algorithms, Speeded-up Robust Features (SURF) [3] and Binary Robust Invariant Scale Key-points (BRISK) [9]. We define four strategies for delegating the tasks of interest point detection and descriptor extraction across the nodes of the VSN, which could potentially allow the VSN resources to be used efficiently and the computational and transmission load to be balanced. Based on a large public image database we present a detailed statistical evaluation of the number, location and scale of SURF and BRISK interest points and descriptors, and use the results of the statistical evaluation to discuss the performance of the delegation strategies.

The statistical characterization of local features has recently received some attention [17], [18]. In [18] the authors evaluate the interest point distribution among different octaves in the case of SURF on a small dataset, while [17] considers the spatial distribution of interest points for various local descriptor algorithms, including SURF but not BRISK, for the particular application of homography estimation and confims our results on this particular aspect of interest point characteristic. Our paper is different from these works as it provides a characterization of a variety of statistical properties of interest points that are relevant for in-network processing is VSNs. While our study is motivated by applications in VSNs [19], [20], [21], our results may provide insight into parallel or distributed processing of visual features on multi-core computing platforms and for stream processing in cloud environments [22], [23]. Furthermore, the evaluation and the comparison of the spatial distributions of the interest points using SURF and BRISK are also relevant for visual tasks such as homography estimation, 3D visualization and tracking, as demonstrated in [17].

The rest of the paper is organized as follows. In Section II we give a background on SURF and BRISK, describe the VSN and the four off-loading strategies. Section III gives statistical analysis of the expected processing load, while SectionsIV-VI evaluates the different off-loading schemes. Finally, in Section VII we discuss the results and conclude our work.

II. Background and System model

We consider a VSN consisting of a single camera node C, a set $\mathcal{P}$, $|\mathcal{P}| = P$ of processing nodes and a data storage node S, as shown in Figure 1a. The processing capability and energy resources of all nodes are limited as well as the communication bandwidth. The transmission channel from C to $\mathcal{P}$ may be broadcast, or may consist of directional links. The goal of the VSN is to compute descriptors for the images captured at C and to transmit them to S.

A. Computing Feature Descriptors

The computation of feature descriptors for an image involves three major processing steps: the detection of the interest points, orientation identification and descriptor extraction.

Interest points are detected by searching for locations with intensity changes in the image. The locations can be blobs or edges and corners. The detector implemented in SURF is an example of a blob detector. SURF calculates blob response maps by convolving the image with the Hessian of second order derivatives of a Gaussian box filter; the response t of a pixel is the determinant of the Hessian matrix. A pixel is an interest point candidate if the *response value* t of the pixel is larger than the detection threshold value Θ. To provide scale invariance, the process is repeated several times, with increasing filter sizes. The filter size is described by the standard deviation σ of the Gaussian distribution. The considered σ parameters are selected according to an octave, octave layer structure. As a final step, responses at the neighboring octave layers are compared to select the interest points and the scale σ of an interest point is determined through interpolation.

The detector of BRISK is an example of a corner detector. A pixel of the image is a corner candidate if 9 consecutive pixels in a circle of 16 pixels around it are darker (or brighter) than the pixel plus (or minus) a threshold Θ. The *response value* t of a pixel is the maximum threshold value for which the pixel is still a corner candidate, and the pixel is an interest point candidate if its response value t exceeds the detection threshold value Θ. BRISK achieves scale invariance by progressively down-sampling the image according to an octave, intra-octave structure, and repeating the detection process for each layer. Finally, interpolation is done between the layers to derive the refined coordinates and the scale σ of the interest point.

To summarize, SURF and BRISK have similar input parameters, that is, the number of octaves and octave layers, respectively intra-octave layers, defining the scale space of interest, and a threshold value Θ. These parameters are selected based on the requirements of the visual task [3]. As a result of the detection process, each detected interest point k in an image i is characterized by its *coordinates* $(x, y)_{i,k}$, the *scale* $\sigma_{i,k}$ and the *response value* $t_{i,k}$.

Detection is followed by orientation identification. This step uses the pixel information within a circular interest area $A_{i,k}$ around the interest point location $(x, y)_{i,k}$. The radius of the interest area is $\sqrt{2}\cdot 10\sigma_{i,k}$ in SURF, while in BRISK the radius is defined by the requirements of the descriptor extraction; in the original implementation the radius is $11.047\sigma_{i,k}$. Figures 1b and 1c show examples of interest points and related interest areas detected by SURF and by BRISK, respectively.

The last step is the extraction of the descriptors $D_{i,k}$. In SURF, descriptor extraction is done using a rectangular interest area $R_{i,k}$ of side length $20\sigma_{i,k}$ within the interest area $A_{i,k}$. In BRISK the interest area $R_{i,k}$ used for extraction is the same as $R_{i,k}$ used for orientation identification. Both SURF and BRISK generate descriptors with a fixed length.

B. Delegation of Feature Computation

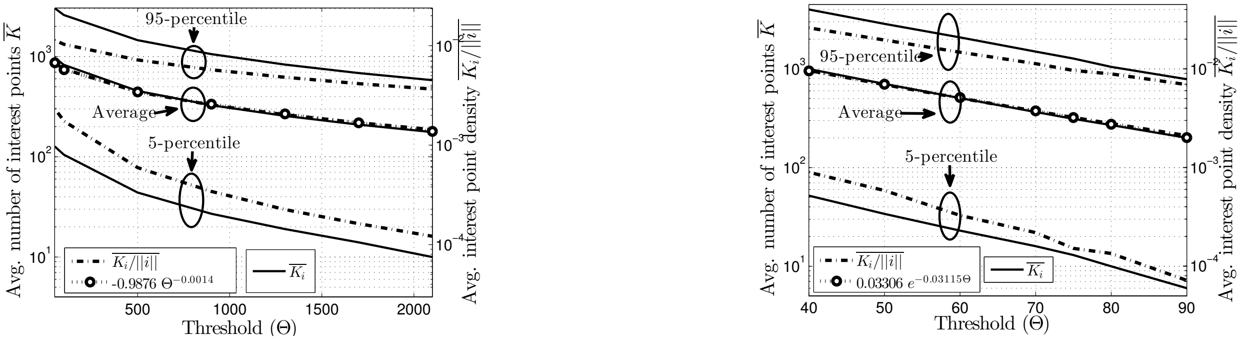
To accelerate the extraction process and to extend the lifetime of the VSN, some or all processing steps can be delegated from the camera node C to $\mathcal{P}$.

The *delegation of the interest point detection* can be done in three ways. C can delegate to a processing node $j \in \mathcal{P}$ an area $Z_{i,j}$ of image i to be processed. We call this *area-split*. The delegated areas need to overlap in order to be able to detect all interest points, and thus the overlaps may need to be transmitted more than once if the wireless links are directional. The width of the overlaps is determined by the size of the largest *interest area*, and is therefore $\sqrt{2} \cdot 10$ and 11.047 times the expected largest scale for SURF and for BRISK, respectively. With *area-split* the number of interest points detected by a processing node depends on the spatial distribution of the interest points. Alternatively, C can assign a processing node the octave parameters (effectively the scales) to be used for interest point detection. We call this *scale-split*. In this case all nodes need to receive the entire image, and the number of interest points detected by each of them depends on the distribution of the interest points across octave layers. Finally, the delegation of interest point detection can be based on both area and scale.

The *delegation of orientation identification* requires the pixel data for $A_{i,k}$ to be transmitted, and the *delegation of descriptor extraction* requires the pixel data for $R_{i,k}$ to be transmitted. The areas for different interest points might overlap, and the overlap can be used to decrease the amount of data transmitted between the nodes for delegating these two processing steps.

The delegation of the processing steps affects the use of the computational and communication resources of the VSN nodes: the data transmission from C to $\mathcal{P}$, the computational load of C and that of the nodes in $\mathcal{P}$, and the data to be transmitted from $\mathcal{P}$ to S. These three are strongly coupled, and therefore the delegation needs to be optimized. We consider the following schemes for delegating processing steps in order to off-load the camera node C:

- *No Detection / No Extraction (ND/NE)*: The camera node performs neither detection nor extraction. Thus, the entire image needs to be sent to the processing nodes, and area-, scale- or hybrid split can be used.
- *Partial Detection / Partial Extraction (PD/PE)*: The camera node detects a subset of the interest points and extracts the related descriptors. The way the subset of interest points to be processed locally at the camera is determined depends on how the processing steps are delegated. In case of *area split*, interest points in an area $Z_{i,0}$ are detected and extracted at the camera. In case of *scale-split* interest points of particular octave-layer parameters are processed locally. The detection and extraction of interest points at large scales would then decrease the redundancy of *area-split*. In case of *hybrid-split* both area and octave-layer could be specified.
- *Complete Detection / No Extraction (CD/NE)*: The camera node detects all interest points and delegates the descriptor extraction to the processing nodes. Given the location and scale of all interest points, the camera node needs only transmit the pixel data for each interest area $A_{i,k}$. If the camera node calculates orientation as well, only the pixel data of $R_{i,k}$ is needed, which, if SURF is used, can decrease the number of pixels to be transmitted by up to one third. At the same time the vector of interest point locations and scales needs to be transmitted. The balancing of the computation among processing nodes can be done based on the known location and scale parameters and becomes trivial if a broadcast link is considered. In the case of directed links, however, the problem is related to graph clustering and is known to be NP-complete.
- *Complete Detection / Partial Extraction (CD/PE)*: The camera node detects all interest points and extracts some of the descriptors. The set of interest points for which descriptors are extracted at the source should be chosen to minimize the remaining image pixels to be transmitted to $\mathcal{P}$. In this paper we consider a simple heuristic,



(a) SURF

(b) BRISK

Fig. 2: Average number of detected interest points and average interest point density vs. detection threshold Θ for (a) SURF and for (b) BRISK.

processing interest points with largest scales at the source, which is optimal when interest areas do not overlap.

Which of these schemes can achieve the best performance depends on two factors. First, on the topology, and the computational and transmission resources of the VSN. Second, on the locations and scales of the interest points, which are unknown a priori. In what follows we address the impact of the latter. We evaluate statistical properties of the interest point and scale-size distribution and assess the expected computational and transmission gains of the above delegation schemes.

C. Evaluation methodology

Our results are based on the 30607 images of the Caltech-256 object category data set [24], processed by the SURF implementation of OpenCV [25] and by the original C++ implementation of BRISK for OpenCV [26]. For SURF we use the default parameters, 4 octaves and 2 octave layers within each octave. For BRISK we consider 4 octave and 4 intra-octave layers. We performed the interest point detection and feature extraction using BRISK and using SURF on a desktop computer, but SURF and BRISK implementations for battery powered sensor platforms with limited memory exist [16], [27], and since the algorithms are deterministic, they would produce the same results on a VSN node.

III. VSN Workload: Number of interest points

As demonstrated in [20], [27], the time needed for interest point detection and for feature extraction are linear functions of the image size $||i||$ (in pixels) and of the number of interest points K_i found in the image, both for SURF and for BRISK, on various hardware platforms. Furthermore, since SURF and BRISK descriptors have a fixed size, the number of interest points also influences the amount of data to be transmitted to the server node S. Therefore we start the evaluation with the distribution of the number of interest points over the set of images.

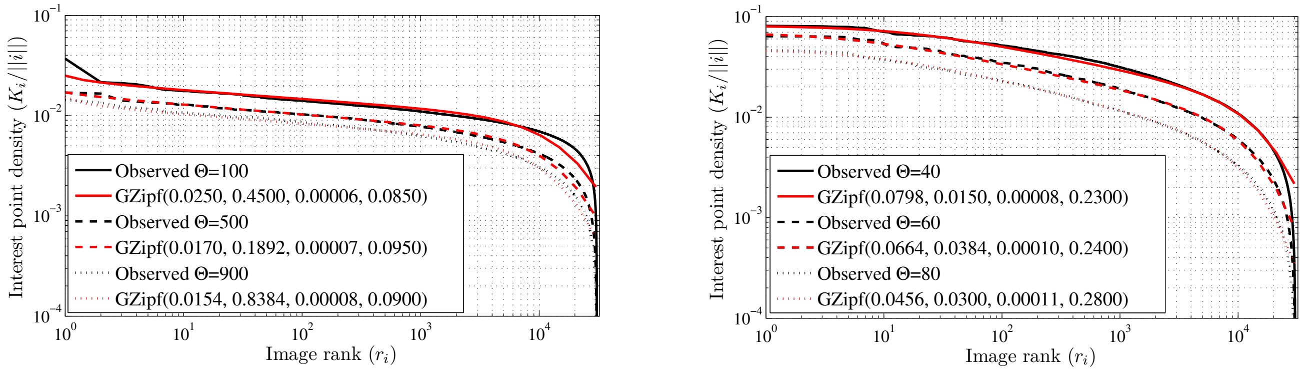
Figure 2a shows the mean number of detected interest points $\bar{K}$ as a function of the detection threshold Θ for SURF on a logarithmic scale. In the considered Θ range, the mean number of interest points decreases by an order of magnitude. The figure shows the best power law fit, $0.9876 \times \Theta^{-0.0014}$, for the curve. The good fit suggests that the decay indeed follows a power law, i.e., is slower than exponential. Thus, the number of interest points decreases fast for low threshold values but with diminishing rate. Therefore, we can conclude that increasing an already high threshold value will not significantly decrease the average computational and transmission load in the VSN.

The figure shows as well the 5- and 95-percentile values, which spread by more than one order of magnitude for any given Θ. Vice versa, to keep the number of interest points constant among the images, the detection threshold has to be adjusted by up to an order of magnitude. To see whether the high variability of the number of interest points can be the result of different image sizes in the data set, we plotted the average interest point density, that is, the average of the number of detected interest points normalized by the image size, $K_i/||i||$, in the same figure. We see similar trends for the interest point density, though with somewhat decreased 5-95-percentile ranges, which confirms that the number of interest points per image varies significantly even for images with the same size.

Figure 2b shows the corresponding results for BRISK. The figure shows the best exponential fit, $0.03306 \times e^{-0.03115\Theta}$, and suggests that the density of BRISK interest points decreases exponentially as a function of the detection threshold in the considered range. Comparing to the results for SURF (cf., Figure 2a), we also see that the variability of the interest point density for given Θ is slightly higher for BRISK.

As the interest point density for a particular threshold would determine the processing load of a VSN, in Figures 3a and 3b we show the rank statistic of the interest point density for three detection threshold values for all images. The considered threshold values are $\Theta = 100, 500, 900$ and $\Theta = 40, 60, 80$ for SURF and for BRISK, respectively, resulting in a range of average number of interest points and interest point densities that allow good retrieval performance based on results in [13], [28].

We observe that the interest point density varies by up to two



(a) SURF

(b) BRISK

Fig. 3: Rank statistics of the interest point density for different detection thresholds and the best-fit generalized Zipf distribution.

orders of magnitude for SURF, and the variability for BRISK is even higher, around three orders of magnitude. Nevertheless, the shape of the curves is similar for the two algorithms and for all Θ values. To understand the characteristics of the curves, we fitted the generalized Zipf law to the data [29]. According to the generalized Zipf law the frequency of an object with rank r is given by

$$f_{GZipf(f_1,\lambda,\mu,\theta)}(r) = \frac{f_1}{\left[1 - \frac{\lambda}{\mu} + \frac{\lambda}{\mu}\, e^{\frac{1}{\theta}\mu r}\right]^{\theta}}. \qquad (1)$$

The generalized Zipf law exhibits an initial plateau, a power-law trunk and an exponential cut-off, and was used to model proportional voting processes and content popularity in the Internet [29], [30]. Figures 3a and 3b show that the fitting is rather accurate, and suggests that there might be a power-law scaling in the number of interest points per image.

Motivated by the above results, in the rest of the paper we use $\Theta = 500$ and $\Theta = 60$ as default threshold values for SURF and for BRISK, respectively. These threshold values lead to similar average interest point densities for the two algorithms, and allow good retrieval performance [13], [28].

We conclude that the number of detected interest points in an image depends significantly on the image content. Therefore, without a-priori information it is difficult to choose detection parameters so as to ensure that an adequate number of interest points is detected. Nevertheless, the processing load due to extracting descriptors can be bounded if it is sufficient to extract descriptors for the M interest points with highest response values only, for some M. We refer to this as the *Top-M* extraction scheme in the following [13]. We denote the set of top-M interest points by $\mathcal{K}_i^M$ and their number by $K_i^M = ||\mathcal{K}_i^M||$. To implement the *Top-$M$* extraction scheme, if the number of detected interest points K_i in image i is larger than M for the detection threshold Θ used, the interest points are ordered according to their response values $t_{i,k}$, and only the M interest points with highest responses are considered for descriptor computation. Nevertheless, if the detection threshold Θ is too high, the number of detected interest points can be less than the target M. Consequently, the number of extracted interest points K_i^M can be less than the target M depending on the threshold Θ used for detection and depending on the image.

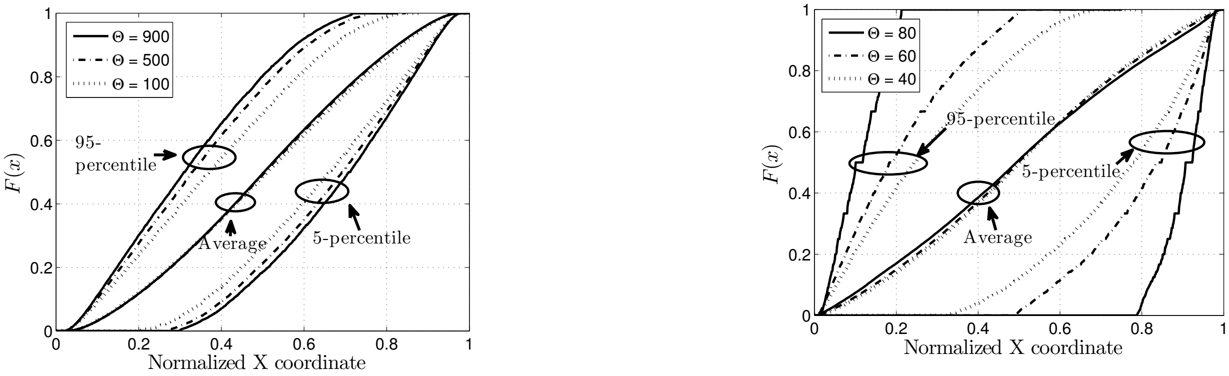
IV. Area-split: Spatial distribution of Interest points

The spatial distribution of the interest points in an image plays an important role in the efficiency of *area-split*, as it affects the distribution of computation and transmission load among the camera node C and the processing nodes $\mathcal{P}$. Under the *ND/NE* and *PD/PE* schemes the spatial distribution is not known by the camera node, hence a large variation of the spatial distribution across images would lead to unpredictable processing load under *area-split*.

A. Spatial distribution of interest points

Let us define $\tilde{x}_{i,k}$ as the normalized X-coordinate of interest point k in image i, i.e., $x_{i,k}$ divided by the width of image i. Furthermore, we define $n_i(\tilde{x})$ to be the number of interest points with normalized X-coordinate less than or equal to $\tilde{x}$ in image i, i.e., $n_i(\tilde{x}) = |\{k|\tilde{x}_{i,k} \leq \tilde{x}\}|$. We define the distribution function for image i as $F_i(\tilde{x}) = \frac{n_i(\tilde{x})}{K_i}$, and the mean over all images $\overline{F}(\tilde{x}) = \frac{1}{N}(\sum_{i=1}^{N} F_i(\tilde{x}))$.

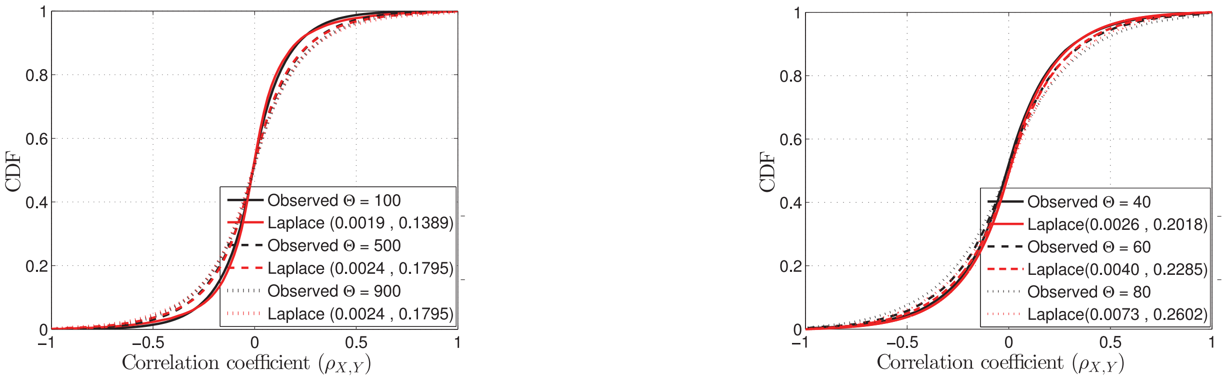
In Figure 4 we show the mean distribution $\overline{F}(\tilde{x})$, and the 5- and 95-percentile values $F_5(\tilde{x})$ and $F_{95}(\tilde{x})$, for different Θ values. Both for SURF and for BRISK, $F_{avg}(\tilde{x})$ is close to the diagonal, which suggests that the spatial distribution of the X-coordinates is close to uniform on average. We see a small deviation close to 0 and close to 1, reflecting that on average fewer interest points can be found close to the image borders, as a consequence of the applied detection techniques. Although the mean is close to uniform, the 5-95-percentile interval is rather wide, particularly for BRISK. The width of the interval increases with Θ, that is, fewer interest points result in a more uneven spatial distribution. As we have seen in Figure 2b, BRISK results in highly varying interest point density under given Θ, and specifically, in the $\Theta = 80$ case gives less then 10 interest points for 5% of the images. This in turn leads to the very wide 5-95-percentile interval for this scenario. The results of the statistical analysis of the Y-coordinate distribution were very similar and are therefore not shown in the paper.



(a) SURF

(b) BRISK

Fig. 4: Spatial distribution of the normalized X-coordinates of the interest points.



(a) SURF

(b) BRISK

Fig. 5: CDF of the correlation coefficient of the normalized X and Y coordinates in the images.

An important question for *area-split* would be whether one could split an image in a number of areas with equal number of interest points if one knew the marginal distributions of the X and Y coordinates of the interest points. To investigate this question we consider the correlation between the X and Y-coordinates. Figure 5 shows the distribution of the correlation coefficients $\rho^i_{X,Y}$ for the X and Y-coordinates over all images i, for SURF and for BRISK, for various threshold values Θ. The figures also show the best fit Laplacian distributions with their location and scale parameter. The location parameter values are close to zero, suggesting that the average $\rho_{X,Y}$ is close to zero both for SURF and for BRISK. The scale parameter of the Laplacian distributions are higher for higher threshold values Θ, suggesting higher correlation when the number of detected interest points is low. In general, the scale parameter of the Laplacian distribution is higher for BRISK, possibly due to images with very low number of interest points. Although the correlation distribution is close to symmetric around zero, the correlation values are non-zero for most images, and thus the interest point coordinates cannot be considered independent. Consequently, knowledge of the marginal distributions would not be sufficient for balancing the load using *area-split*.

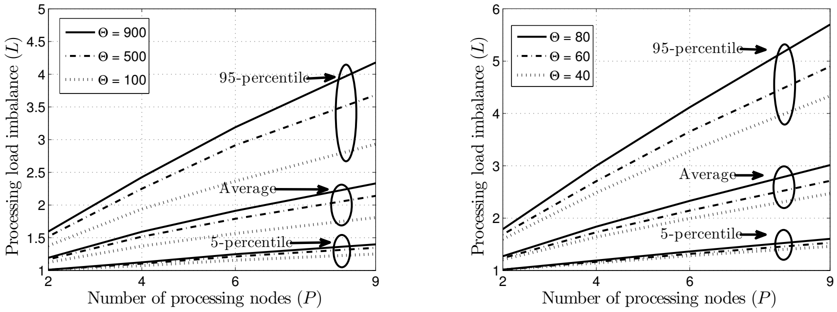
B. Processing load imbalance

Let us now turn to the impact of the spatial distribution of interest points on the feasibility of balancing the processing load under *area-split*. We divide each image into P equal-sized areas, for $P = 2, 4, 6, 9$, using vertical and horizontal cuts, and assign area $Z_{i,j}$ to processing node j. We denote by $n_{i,j}$ the number of interest points detected by node j. Since the interest point distribution is close to uniform on average (cf., Figure 4), the total computation and transmission load among the processing nodes will be nearly balanced after processing many different images. Nevertheless, over the course of processing a few images, the non-uniform spatial distribution and the correlation of the X and Y coordinates may lead to unbalanced processing loads, and thus highly varying processing times. This may be problematic if the processing has to be done under delay constraints.

We define the processing load imbalance L_i for an image i as the ratio of the maximum number of interest points in any of the areas divided by the average of number of interest points per area,

$$L_i = \frac{\max\limits_{1\leq j\leq P} n_{i,j}}{K_i/P}. \qquad (2)$$

By definition $1 \leq L_i \leq P$. A value close to 1 corresponds to balanced load distribution, while a value of $L = P$ means that



(a) SURF

(b) BRISK

Fig. 6: Processing load imbalance as a function of the number of processing nodes.

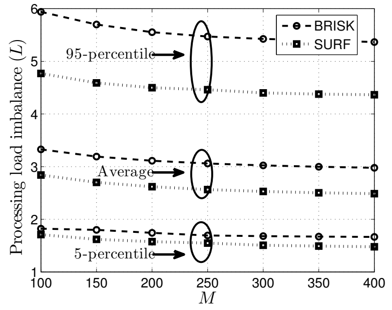
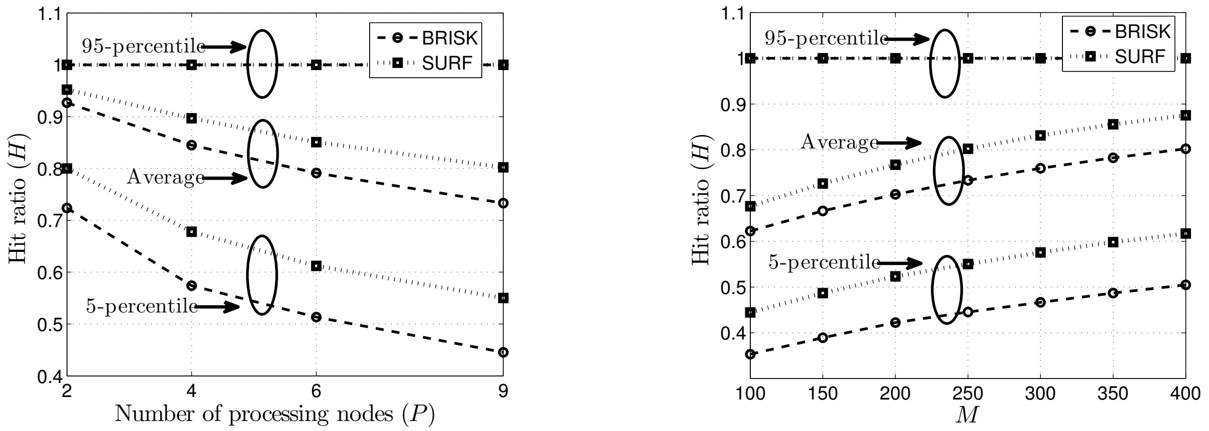


Fig. 7: Processing load imbalance vs. number of interest points M to be extracted for P = 9.



(a)

(b)

Fig. 8: Hit ratio vs. (a) number of nodes P for M = 250, (b) number of interest points M to be extracted for P = 9.

all interest points fall into the same area and are processed by a single processing node.

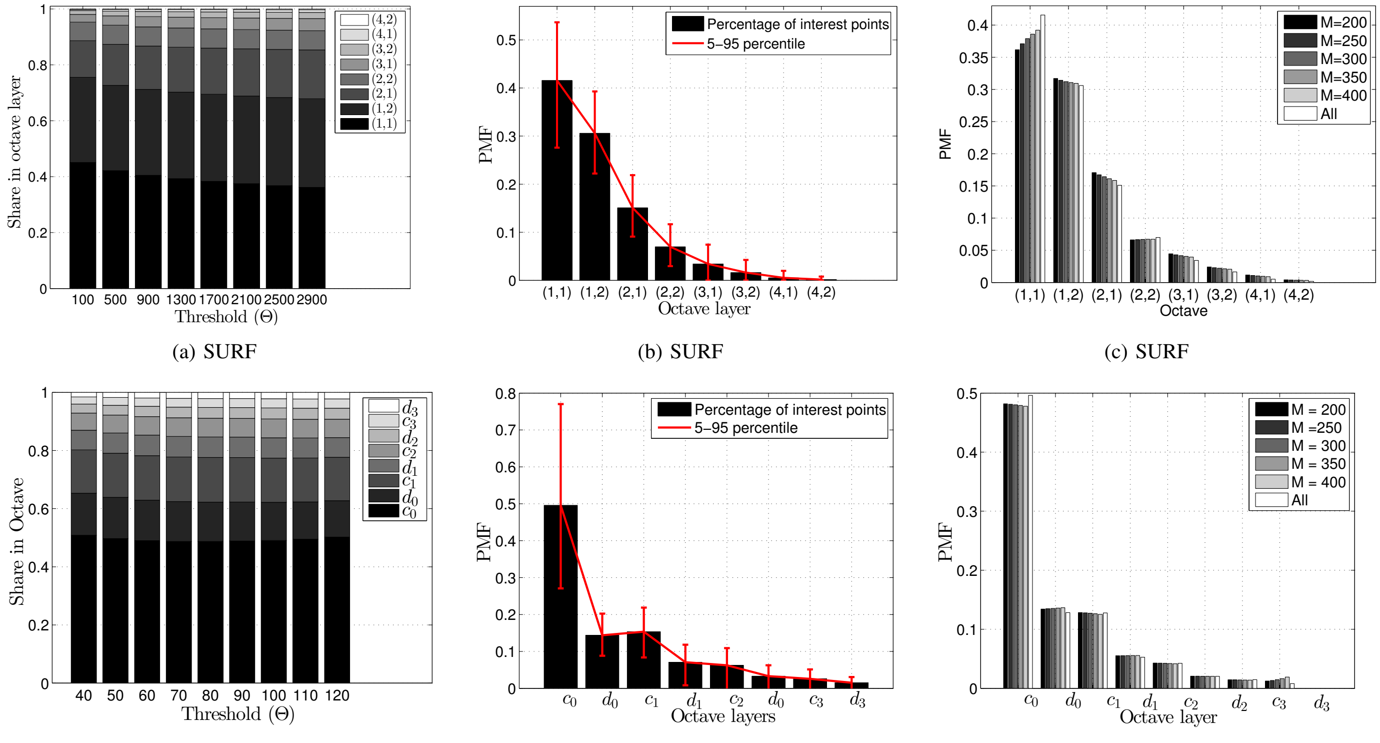
Figure 6 shows the average and the 5-95-percentiles of the load imbalance L_i as a function of the number of processing nodes P for various Θ values, for SURF and for BRISK. The imbalance increases nearly linearly for the considered P values; the increase is faster for larger Θ, and is generally slower for SURF, which can be explained based on the results for the spatial distribution shown in Figure 4. The results show that on average the most loaded processing node would have close to three times more workload than the average value, and the imbalance exceeds 4 and 5 for 5% of the images for SURF and for BRISK, respectively.

Next, we consider the imbalance under the Top-M extraction scheme, for two cases. First, we consider the ideal case that the processing nodes $\mathcal{P}$ are aware of the threshold Θ that has to be used to detect M interest points in total.

Figure 7 shows the average and the 5-95-percentiles of the processing load imbalance L for $P = 9$ for SURF and for BRISK, for values of M that provide good retrieval performance [13], [28]. The imbalance decreases as the number M of interest points to be detected and extracted increases, but comparing the results with the ones in Figure 6, extracting the Top-M interest points leads to increased processing load imbalance, even at $M = 400$.

Second, we consider the practical case that the processing nodes $\mathcal{P}$ do not know the correct threshold Θ that should be used to detect the Top-M interest points. Therefore, each processing node $j \in \mathcal{P}$ has to determine which of the interest points detected in its area $Z_{i,j}$ belong to the Top-M interest points in the entire image. Without a-priori information, the most processing intensive approach would be to have processing node j detect and extract up to M interest points in its area $Z_{i,j}$. In the worst case this approach would result in $(P-1)M$ interest points to be detected and extracted unnecessarily. The least processing intensive approach, which we consider, is that processing node j extracts up to M/P interest points, the interest points with highest response in $Z_{i,j}$. Let us denote the set of extracted interest points by $\kappa^M_{i,j}$. If we denote by $n_{i,j}$ the number of interest points detected in area $Z_{i,j}$ of image i then the number of extracted interest points at processing node j is $|\kappa^M_{i,j}| = \min\{n_{i,j}, M/P\}$.

While the processing load is expected to be balanced in this case, without cooperation the processing nodes may not be able to find all top-M interest points, which may lead to decreased visual analysis performance. To quantify the ability of the nodes to extract the Top-M interest points we define



(a) SURF (b) SURF (c) SURF

(d) BRISK (e) BRISK (f) BRISK

Fig. 9: (a,d) Share of interest points per octave vs. detection threshold. (b,e) Interest point distribution for SURF $\Theta = 500$ and BRISK $\Theta = 60$. Mean, 5 and 95 percentiles. (c,f) Distribution of top M interest points per octave layer.

the *hit ratio* as the ratio of the number of extracted Top-M interest points and the number K_i^M of Top-M interest points that should have been extracted

$$H_i^M = \frac{||\bigcup_{j=1}^{P} \kappa_{i,j}^M \bigcap \mathcal{K}_i^M||}{K_i^M}. \tag{3}$$

The hit ratio is by definition bounded by $\frac{1}{P} \leq H_i^M \leq 1$. $H_i^M = 1$ when all $K_i^M = ||\mathcal{K}_i^M||$ top-M interest points are processed, and $H_i^M = \frac{1}{P}$ when one area contains all the top-M interest points.

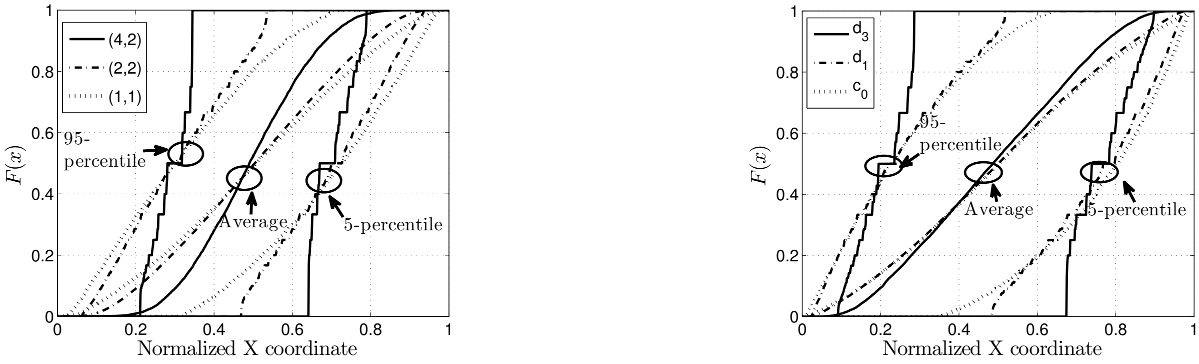
Figure 8a shows the average and the 5-95-percentiles of the hit ratio H_i^M as a function of the number of processing nodes P, for $M = 250$. When there are many processing nodes, only about 70% of the top interest points are detected, and for 5% of the images this ratio drops below 50%, which may deteriorate visual analysis performance. Consistent with the less even spatial distribution of interest points, the hit ratios are lower for BRISK. Figure 8b shows the results as a function of M for $P = 9$. The figure shows that the hit ratio increases with M, because the distribution of the Top-M interest points among areas becomes more balanced, but the 5 percentiles are still very low.

From the above results we conclude that without a-priori information on the interest point distribution, e.g., in the case of *ND/NE* and *PD/PE*, the processing load imbalance can be very high. This in turn can lead to significantly increased processing times in the VSN, as well as to decreased network lifetime due to the uneven energy consumption at the processing nodes $\mathcal{P}$. Similarly, the Top-M extraction scheme either requires knowledge of the detection threshold Θ to be used, or if that is not known a-priori, then each processing node either has to extract more than M/P interest points in its area, which leads to processing overhead, or the processing nodes need to coordinate to agree on the set of Top-M interest points, which leads to communication overhead and delay.

Area-split could become an efficient solution to delegate the feature extraction tasks to the processing nodes if the camera node could obtain the location of the interest points by performing complete detection, i.e., *CD/NE* and *CD/PE*. Alternatively, the camera node could try to predict the interest point distribution if the subsequent images are correlated, as, for example, in the case of visual analysis of video sequences.

V. Scale-split: Octave distribution of interest points

We now consider the distribution of the interest points over the octave layers they are detected at. This distribution is important for understanding the feasibility of *scale-split*, i.e., when each processing node performs detection only in a part



(a) SURF

(b) BRISK

Fig. 10: Spatial distribution of interest points along the X-axis for different octave layers.

of the scale space.

We again start with considering the *ND/NE* scheme, thus the delegation needs to be based on the parameters of the detection algorithm, which are the octave layers in the scale space. Therefore, we evaluate the interest point distribution as a function of this parameter. For SURF we consider the default parameters of 4 octaves and 2 octave layers within each octave, denoted as $(1,1),(1,2),(2,1)\ldots(4,2)$. For BRISK we consider 4 octave as well as intra-octave layers, denoted as $c_0, d_0, c_1 \ldots d_3$. Thus, for both algorithms, the interest point detection is performed in 8 rounds.

Figures 9a and 9d show the mean value of the ratio of interest points detected at the different octave layers at various detection threshold values for SURF and for BRISK, respectively. We note that the majority of the interest points are detected at low octaves/layers, representing a small scale σ: at octave/layer $(1,1)$ and $(1,2)$ under SURF and at c_0 under BRISK. The share of interest points detected at low octaves/layers decreases slightly as Θ is increased. Extreme cases for SURF were reported in [18] for very high threshold values and for a limited set of images.

Figures 9b and Figure 9e show the average and the 5-95-percentile of the probability mass function (PMF) of the interest point distribution across the octave layers for SURF and for BRISK, respectively. The distributions are very skewed; for example, processing interest points at the lowest octave layer results in 40% and 50% of the processing load for SURF and for BRISK, respectively, while the two highest octave layers together would result in 0.3% and in 5% of the processing load. The 5-95-percentile intervals are wide for both algorithms, and even here the BRISK results show higher variability across the images.

Figures 9c and 9f show that the octave layer distribution remains skewed even under the Top-M detection scheme. The distributions are rather similar for all M values, apart from the $(1,1)$ layer results for SURF, which suggests that interest points with very high responses are slightly less likely to be found in the lowest octave layer.

An important consequence of the skewed octave layer distribution is that delegation using *scale-split* only is not very suitable for balancing the load under *ND/NE* and under *PD/PE*, if balancing the processing load is desired, e.g., for every individual image. We therefore now consider combining *scale-split* with *area-split* in order to balance the processing load at low octave layers among the processing nodes. To evaluate the benefit of combining *scale-split* with *area-split*, Figure 10 shows the average cumulative distribution function $F_{avg}(\tilde{x})$ of the normalized X-coordinates, and the 5-95-percentiles for interest points detected at three different octave layers. For comparison we show a low, a medium and a high scale octave layer for both algorithms: layers $(1,1),(2,2)$ and $(4,2)$ for SURF and layers c_0, d_1 and d_3 for BRISK. The results for low octave layers are very similar to the ones in Figure 4. For higher octave layers the distribution of the average over all images gets further from uniform and the 5-95-percentile intervals get wider, since interest points with large scale σ_k are not detected close to the borders of the images, and their number is low in general.

From these results we can conclude that *scale-split*, even when combined with *area-split* would lead to extremely unbalanced load under *ND/NE*, and does not seem to have any advantage compared to pure *area-split*. The question that remains is whether letting the camera node detect and potentially extract at high octave layers, combined with *area-split* among the processing nodes $\mathcal{P}$ could make *scale-split* beneficial. This will be evaluated in the next section.

VI. Involving the camera node

Considering the statistics of the scale and the spatial distribution of the interest points, we see that both show high variability across the images. Thus, under *ND/NE* one either needs to be able to predict the detection parameters that lead to a balanced processing load or one needs extensive coordination between the processing nodes. We therefore now consider the *PD/PE*, the *CD/NE* and the *CD/PE* schemes. These schemes provide the camera node with information that it could potentially use for controlling the balance of the delegation of the descriptor extraction.

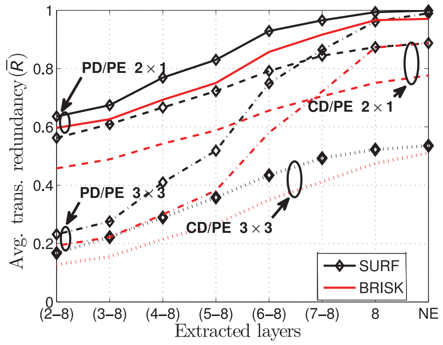
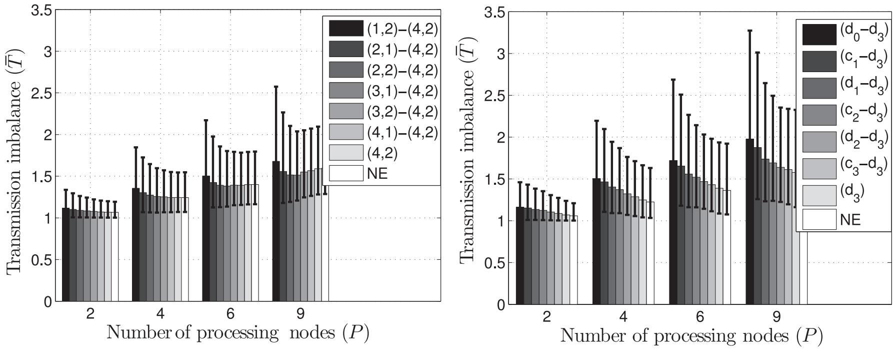


Fig. 11: Average transmission redundancy under area-split for *PD/PE* and *CD/PE*.



(a) SURF

(b) BRISK

Fig. 12: *CD/NE* and *CD/PE* transmission imbalance for (a) SURF and (b) BRISK.

A. Transmission redundancy and imbalance

We start with considering the transmission of the image pixel information from C to $\mathcal{P}$. If a broadcast link is used, all pixel information is transmitted to all nodes at once, and offloading incurs no redundant transmission. Nevertheless, in the case of directed links the amount of data to be transmitted to $\mathcal{P}$ depends on the offloading scheme, and certain areas of the image may have to be transmitted several times.

For the evaluation we consider *area-split* with $P = 2, 4, 6$ and 9 processing nodes. The subareas assigned to the nodes are constructed as follows. First, the image is cut with horizontal and vertical cuts into 2×1, 2×2, 3×1, or 3×3 rectangular areas. Then, overlapping areas are added to ensure that the interest points close to the cuts can be detected. For *ND/NE* and *PD/PE* the rectangular areas are of equal size and the width of the overlap is defined by the radius of the interest area $A_{i,k}$ at the highest possible scale σ. For *CD/NE* and *CD/PE* we optimize the location of the cuts so that each area includes an equal number of interest points, and the overlaps depend on the actual interest point positions $(x, y)_{i,k}$ and scales $\sigma_{i,k}$.

We define the *transmission redundancy* as the sum of the sizes of the transmitted areas divided by the image size $||i||$ and normalized by the number of areas

$$R_i = \frac{\sum_{j=1}^{P} ||Z_{i,j}||}{||i||P}. \tag{4}$$

The transmission redundancy is by definition $1/P \leq R_i \leq 1$. $R_i = 1$ means that every processing node needs to receive the entire image, while $R_i = 1/P$ means no redundancy.

Figure 11 shows the average transmission redundancy for SURF and for BRISK under the *PD/PE* and the *CD/PE* schemes. We show the results for decreasing levels of extraction at the camera node. The leftmost axis label $(2-8)$ means that all but the lowest octave layer are extracted at C, the axis label 8 means that only the highest octave layer is extracted at C, while *NE* stays for no extraction at the camera node, which corresponds to *PD/NE* or *CD/NE*. The figure shows results for $P = 2$ and $P = 9$, that is, 2×1 and 3×3 cuts.

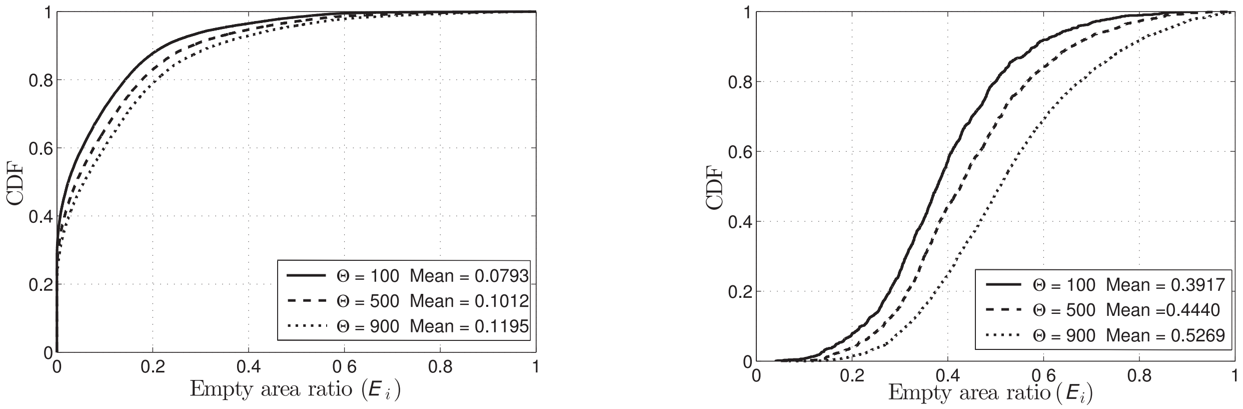
The redundancy is close to 1 under *PD/NE* and *CD/NE*, both for SURF and for BRISK, though the rendundancy for BRISK is in general lower, because the interest areas $A_{i,k}$ are smaller. Interestingly, the redundancy decreases quite fast and steadily as more and more octave layers are extracted at C. Thus, there is a significant trade-off between the transmission redundancy and the processing load at C. A similar trade-off can be observed when comparing the redundancy for *PD/PE* and for *CD/PE*. *CD/PE* allows the overlaps to be minimized, which helps to decrease the redundancy compared to *PD/PE*, at the price of increased processing at C. We conclude that detection at the camera node can significantly decrease the redundancy in the data transmitted to the processing nodes.

In principle *CD/NE* and *CD/PE* allow one to balance the processing load among the processing nodes $\mathcal{P}$ as the interest point locations are known in the camera node, and they would even allow a lower transmission redundancy in the case of directed links. Balancing the processing load would, however, require that the different processing nodes process areas of different sizes.

To quantify the imbalance in terms of the area sizes needed to balance the processing load, we define the *transmission imbalance* as the ratio of the size of the largest area transmitted to a node divided by the average of the area sizes

$$T_i = \frac{\max_{1 \leq j \leq P} ||Z_{i,j}||}{\sum_{j=1}^{P} ||Z_{i,j}||/P}. \tag{5}$$

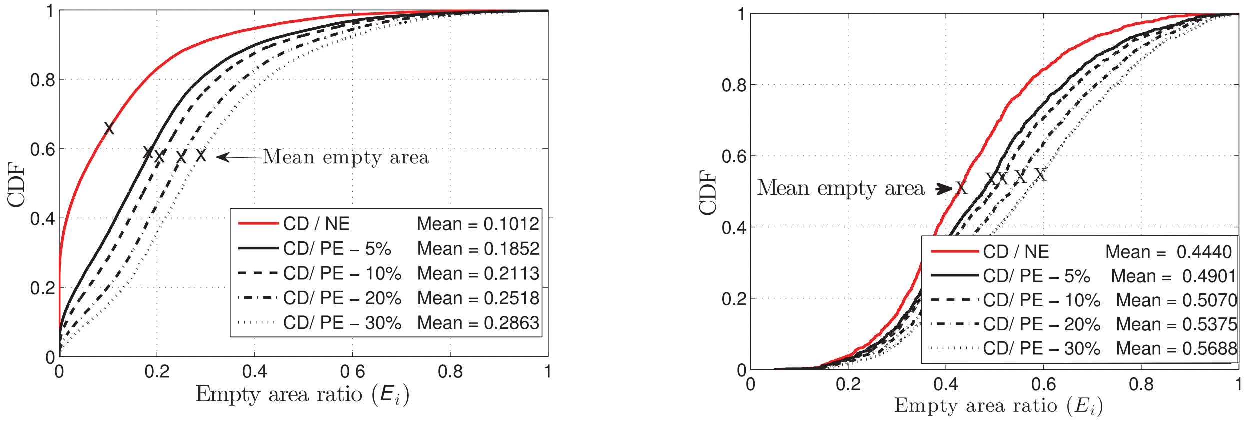
As the size of the transmitted areas depends on the layers extracted at C, Figures 12a and 12b show the transmission imbalance for SURF and for BRISK with different levels of extraction at the camera node and as a function of the number P of processing nodes. For SURF, the average tranmission imbalance $\overline{T}$ is only little affected by the number of octave layers extracted at C, but the 5-95-percentiles are slightly wider when only the lowest layers are processed at $\mathcal{P}$. The number of processing nodes has a minor impact on the average imbalance, although the 5-95-percentile interval gets wider when there are more nodes. For BRISK these tendencies



(a) SURF

(b) BRISK

Fig. 13: Empty area distribution under *CD/NE* for various detection thresholds, for (a) SURF and for (b) BRISK.



(a) SURF

(b) BRISK

Fig. 14: Empty area ratio under *CD/NE* and *CD/PE* for (a) SURF and for (b) BRISK.

are more pronounced, and the average and percentile values are generally higher, which is a consequence of the higher variability of the interest point locations for BRISK (cf. Fig. 4).

Based on these results we conclude that partial extraction at the camera node is an efficient solution for decreasing the amount of redundant data transmission, at the price of increased computational load at the camera node. At the same time, if interest point detection is done at the camera node to aid balancing the load of the processing nodes then, interestingly, partial extraction at the camare node does not always help to balance the amount of data to be transmitted to the individual nodes.

B. Empty area distribution

Let us finally consider *CD/NE* and *CD/PE*, that is, when the camera node C performs the detection of *all* interest points. As in this case the location and the scale of the interest points are known, it could be sufficient to transmit only the interest areas around the interest points to the processing nodes $\mathcal{P}$, that is, a disc with radius $\sqrt{2} \cdot 10\sigma_{i,k}$ for each interest point k in image i. We refer to the area of an image not covered by any interest area as *empty area*, and we define the *empty area ratio* as the size of the empty area normalized by the image size $||i||$,

$$E_i = 1 - \frac{||\bigcup_{k=1}^{K} A_{i,k}||}{||i||}. \tag{6}$$

Intuitively, the higher the empty area ratio the less data need to be transmitted from the camera node.

1) Empty area distribution under CD/NE*:* Figures 13a and 13b show the distribution of the empty area ratio E_i for SURF and for BRISK for *CD/NE*, and for various detection threshold values Θ.

The ratio of the empty area turns out to be very small for SURF. It increases slightly with increased detection threshold because less interest points are detected. For the default $\Theta = 500$, the average achievable gain of not transmitting empty areas from the camera node to the processing nodes is roughly 10%, but around 30% of the images do not have empty area at all. Thus, *CD/NE* might not significantly reduce the amount of data to be transmitted from the camera node to the processing nodes for SURF. BRISK leads to significantly larger empty area ratios; on average 40-50% of the images is empty area. The higher empty area ratios are due to two reasons. First, the radii of the interest areas are in general smaller for BRISK. Second, the interest points happen to be more clustered because of the type of detector.

2) Empty area distribution under CD/PE*:* A natural way to increase the empty area ratio is to perform partial feature extraction at the camera node, that is, *CD/PE*, as the interest areas that correspond to interest points processed at the camera node C do not need to be transmitted. In principle the set of interest points extracted at the camera node could be chosen such as to maximize the empty area, but doing so requires combinatorial optimization due to the overlaps between interest areas. We instead consider a low complexity heuristic to choose the interest points to be extracted at C: for a given percentage of the interest points, starting from the highest scale, descriptors are extracted at C. Figures 14a and 14b show the CDF of the empty area ratio E_i under *CD/NE* and *CD/PE* when 5% to 30% of the feature descriptors are extracted at the camera node. Extracting 10% of the descriptors at C for SURF doubles the empty area on average, and the probability that an image does not have empty area at all gets close to zero, which is a significant gain. The gain of partial extraction is much lower for BRISK, because the empty area ratios E_i are already relatively high without partial extraction.

We conclude, that for BRISK not transmitting the empty areas seems to be an efficient solution to decrease the transmission cost from C to $\mathcal{P}$, even without local extraction at C. The same solution gives significantly lower gain for SURF, for which partial extraction at the camera node is needed to increase the mean empty area ratio above 10%.

VII. Conclusion

We provided a statistical characterization of SURF and BRISK interest points based on a large public image database with the aim of getting insight into the efficiency of different design choices for the delegation of processing tasks in a visual sensor network. Motivated by the limited communication, processing and energy resources of the nodes, we evaluated whether it is possible to delegate the processing tasks so that the transmission load in the network is balanced and possibly decreased, and the processing load is balanced across the nodes.

Our results show that the interest point density, the spatial distribution of the interest point locations, and the octave layer distribution of the interest points vary significantly between images. These properties hold both for SURF and for BRISK, and are expected to hold for all similar, blob or corner detection based local feature extraction algorithms. As a consequence, a balanced allocation of the load among processing nodes is infeasible without a-priori information on the image characteristics or without coordination between the processing nodes. A-priori information could be obtained through leveraging potential temporal correlation between subsequent images in the case of video, or it can be obtained through interest point detection performed at the camera node. Our results show that the transmission redundancy needed for *area-split* can be significantly decreased through octave-based partial detection and partial extraction at the camera node. Complete interest point detection at the camera allows balancing the computational load of the processing nodes at the price of increased computational load at the camera node, but can significantly decrease the transmission load for BRISK, as significant parts of the image do not have to be transmitted. Comparing SURF with BRISK our results show that a-priori information is more important for BRISK, as the spatial distribution of interest points varies more. At the same time, if a-priori information is available through detection at the camera node, the gain in terms of decreased transmission load is more substantial when BRISK is used.

VIII. Acknowledgments

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 No.: 296676.

References

[1] A. Marcus and O. Marques, “An eye on visual sensor networks,” *IEEE Potentials*, vol. 31, no. 2, pp. 38 –43, Apr 2012.

[2] D. Lowe, “Object recognition from local scale-invariant features,” in *Proc. of IEEE ICCV*, 1999, pp. 1150–1157.

[3] H. Bay, A. Ess, T. Tuytelaars, and L. V. Gool, “Speeded-up robust features (SURF),” *Computer Vision and Image Understanding*, vol. 110, no. 3, pp. 346 – 359, 2008.

[4] M. Calonder, V. Lepetit, C. Strecha, and P. Fua, “BRIEF: Binary robust independent elementary features,” in *Proc. of ECCV*, 2010.

[5] S. Leutenegger, M. Chli, and R. Siegwart, “BRISK: Binary robust invariant scalable keypoints,” in *Proc. of IEEE ICCV*, 2011.

[6] L.-Y. Duan, X. Liu, J. Chen, T. Huang, and W. Gao, “Optimizing JPEG quantization table for low bit rate mobile visual search,” in *IEEE Visual Communications and Image Processing (VCIP)*, 2012.

[7] J. Chao, H. Chen, and E. Steinbach, “On the design of a novel JPEG quantization table for improved feature detection performance,” in *Proc. of IEEE Intl. Conf. on Image Processing (ICIP)*, 2013.

[8] E. Rosten, R. Porter, and T. Drummond, “Faster and better: A machine learning approach to corner detection,” *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 32, no. 1, pp. 105–119, 2010.

[9] M. C. Stefan Leutenegger and R. Y. Siegwart, “Brisk: Binary robust invariant scalable keypoints,” in *Proc. of IEEE Intl. Conf. on Computer Vision (ICCV)*, 2011.

[10] H. Jegou, F. Perronnin, M. Douze, J. Sanchez, P. Perez, and C. Schmid, “Aggregating local image descriptors into compact codes,” *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 34, no. 9, pp. 1704–1716, 2012.

[11] A. Redondi, L. Baroffio, J. Ascenso, M. Cesana, and M. Tagliasacchi, “Rate accuracy optimization for binary descriptors,” in *Proc. of IEEE Intl. Conf. on Image Processing (ICIP)*, 2013.

[12] V. R. Chandrasekhar, S. S. Tsai, G. Takacs, D. M. Chen, N.-M. Cheung, Y. Reznik, R. Vedantham, R. Grzeszczuk, and B. Girod, “Low latency image retrieval with progressive transmission of chog descriptors,” in *Proc. of ACM Multimedia workshop on Mobile cloud media computing*, 2010.

[13] A. Redondi, M. Cesana, and M. Tagliasacchi, “Rate-accuracy optimization in visual wireless sensor networks,” in *Proc. of IEEE ICIP*, 2012.

[14] D.-N. Ta, W.-C. Chen, N. Gelfand, and K. Pulli, “SURFTrac: Efficient tracking and continuous object recognition using local feature descriptors,” in *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2009.

[15] L. Baroffio, M. Cesana, A. Redondi, S. Tubaro, and M. Tagliasacchi, “Coding video sequences for visual features,” in *Proc. of IEEE Intl. Conf. on Image Processing (ICIP)*, 2013.

[16] L. Baroffio, A. Canclini, M. Cesana, A. Redondi, M. Tagliasacchi, G. Dán, E. Eriksson, V. Fodor, J. Ascenso, and P. Monteiro, "Enabling visual analysis in wireless sensor networks," in *Proc. of IEEE Intl. Conf. on Image Procesing (ICIP), Show and Tell*, October 2014.

[17] E. Bostanci, N. Kanwal, and A. F. Clark, "Spatial statistics of image features for performance comparison," *IEEE Transactions on Image Processing*, vol. 23, no. 1, Jan 2014.

[18] S. Ehsan, N. Kanwal, E. Bostanci, A. F. Clark, and K. D. McDonald-Maier, "Analysis of interest point distribution in SURF octaves," in *Proc. of Intl. Conf. on Machine Vision (ICMV 2010)*, Apr 2010.

[19] E. Eriksson, G. Dán, and V. Fodor, "Prediction-based load control and balancing for feature extraction in visual sensor networks," in *Proc. of IEEE Intl. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, May 2014.

[20] ——, "Real-time distributed visual feature extraction from video in sensor networks," in *Proc. of IEEE Intl. Conf. on Distributed Computing in Sensor Systems (DCOSS)*, May 2014.

[21] A. Redondi, M. Cesana, M. Tagliasacchi, I. Filippini, G. Dán, and V. Fodor, "Cooperative image analysis in visual sensor networks," *Elsevier Ad-hoc Networks*, 2015.

[22] Z. Fang, D. Yang, W. Zhang, H. Chen, and B. Zang, "A comprehensive analysis and parallelization of an image retrieval algorithm," in *IEEE International Symposium on Performance Analysis of Systems and Software (ISPASS)*, April 2011, pp. 154–164.

[23] B. Veeravalli and S. Ranganath, "Theoretical and experimental study on large size image processing applications using divisible load paradigm on distributed bus networks," *Image and Vision Computing*, vol. 20, no. 1314, pp. 917 – 935, 2002.

[24] G. Griffin, A. Holub, and P. Perona, "Caltech-256 object category dataset," 2007. [Online]. Available: http://resolver.caltech.edu/CaltechAUTHORS:CNS-TR-2007-001

[25] "OpenCV." [Online]. Available: http://opencv.org/

[26] "Brisk Implementation." [Online]. Available: http://www.asl.ethz.ch/people/lestefan/personal/BRISK

[27] A. Redondi, L. Baroffio, A. Canclini, M. Cesana, and M. Tagliasacchi, "A visual sensor network for object recognition: Testbed realization," in *Proc. of IEEE/Eurasip Intl. Conf. on Digital Signal Processing (DSP)*, July 2013.

[28] G. Takacs, V. Chandrasekhar, S. Tsai, D. Chen, R. Grzeszczuk, and B. Girod, "Rotation-invariant fast features for large-scale recognition and real-time tracking," *Signal Processing: Image Communication*, vol. 28, pp. 334–344, 2013.

[29] G. Dán and N. Carlsson, "Power-law revisited: A large scale measurement study of P2P content popularity," in *Proc. of Intl. Workshop on Peer-to-Peer Systems (IPTPS '10)*, Apr 2010.

[30] M. L. Lyra, U. M. S. Costa, R. N. C. Filho, and J. S. Andrade, "Generalized Zipf's law in proportional voting processes," *Europhysics letters*, vol. 62, no. 1, pp. 131–137, Apr 2003.

György Dán (M'07) is an associate professor at KTH Royal Institute of Technology, Stockholm, Sweden. He received the M.Sc. in computer engineering from the Budapest University of Technology and Economics, Hungary in 1999, the M.Sc. in business administration from the Corvinus University of Budapest, Hungary in 2003, and the Ph.D. in Telecommunications from KTH in 2006. He worked as a consultant in the field of access networks, streaming media and videoconferencing 1999-2001. He was a visiting researcher at the Swedish Institute of Computer Science in 2008, a Fulbright research scholar at University of Illinois at Urbana-Champaign in 2012-2013, and an invited professor at EPFL in 2014-2015. His research interests include the design and analysis of content management and computing systems, game theoretical models of networked systems, and cyber-physical system security in power systems.

Muhammad Altamash Khan received his B.E. in Electrical Engineering from the National University of Sciences and Technology (NUST), Pakistan in 2006 and his M.Sc. in Wireless Systems from KTH Royal Institute of Technology, Sweden in 2011. He worked as Research engineer at the Lab for Automatic Control and at the Lab for Communication Networks from 2011 to 2013. He worked on the statistical characterization of image/video feature extraction, implementation and testing of motion control for networked robots, and implementing the routing & MAC protocols for wireless sensor networks. He has been working as a Marie Curie fellow since April 2014 at FKIE Fraunhofer, Wachtberg Germany. His research interests include non-linear state estimation methods, multi-target tracking and wireless sensors networks.

Viktoria Fodor (M03) received the M.Sc. and Ph.D. degrees from the Budapest University of Technology and Economics, Budapest, Hungary, in 1992 and 1999, respectively, both in computer engineering. In 1994 and 1995, she was a Visiting Researcher with Polytechnic University of Turin, Turin, Italy, and with Boston University, Boston, MA. In 1998, she was a Senior Researcher with the Hungarian Telecommunication Company. Since 1999, she has been with the KTH Royal Institute of Technology, Stockholm, Sweden, where she now acts as an Associate Professor with the Laboratory for Communication Networks. Her current research interests include network performance evaluation, cognitive and cooperative communication, protocol design for sensor and multimedia networking.