Feature Matching Performance of Compact Descriptors for Visual Search

Vijay Chandrasekhar[1,2], Gabriel Takacs[1], David M. Chen[1],
Sam S. Tsai[1], Mina Makar[1], Bernd Girod[1]
[1]*Information System Labs, Stanford University, CA*
[2]*Institute for Infocomm Research, Singapore*

Abstract

MPEG is currently developing a standard titled Compact Descriptors for Visual Search (CDVS) for descriptor extraction and compression. In this work, we report comprehensive patch-level experiments for a direct comparison of low bitrate descriptors for visual search. For evaluating different compression schemes, we propose a data set of matching pairs of image patches from the MPEG-CDVS image-level data sets. We propose a greedy rate allocation scheme for distributing bits across different spatial bins of the SIFT descriptor. We study a scheme based on Entropy Constrained Vector Quantization and greedy rate allocation, which performs close to the performance bound for any compression scheme. Finally, we present extensive feature-level Receiver Operating Characteristic (ROC) comparisons for different compression schemes (Vector Quantization, Transform Coding, Lattice Coding) proposed during the MPEG-CDVS standardization process.

1. Introduction

For mobile visual search applications, the size of data sent over the network needs to be as small as possible to reduce latency and improve user experience. One approach to the problem is to transmit the JPEG compressed query image over the network, but this might be prohibitively expensive at low uplink speeds. An alternate approach to sending JPEG images is to extract feature descriptors on the mobile device, compress the descriptors and transmit them over the network. Such an approach has been demonstrated to reduce the amount of transmission data significantly [1]. To this end, MPEG is currently developing a standard titled Compact Descriptors for Visual Search (CDVS) for descriptor extraction and compression [2], [3].

The evaluation framework for selection of algorithms [2] consists of a comprehensive set of pairwise image matching and retrieval experiments. At varying rates between 0.5 KB to 16 KB, proponents of algorithms are required to report True Positive Rate (TPR), subject to a False Positive Rate (FPR) of 1%. For retrieval experiments from a database of 1 million images, proponents are required to report Mean Average Precision (MAP) and Precision at 1 at different rate points. However, under the constraints of the evaluation framework, there are several parameters that can be optimized for a proposed scheme: the number of bits per descriptor, the number of descriptors per image, pairwise-matching distance threshold and ratio-threshold parameters, the size of codebook in a Bag-of-Words retrieval framework [4], the number of images in the Geometric Consistency Check (GCC) list, etc.

Here, we consider an experimental set up for a direct comparison of low bitrate descriptors. We create a data set of matching and non-matching pairs of patches from the MPEG-CDVS data sets, similar to the popular patch-level data sets *Liberty*, *Trevis Fountain* and *Notredame*

1068-0314/14 $31.00 © 2014 IEEE
DOI 10.1109/DCC.2014.50

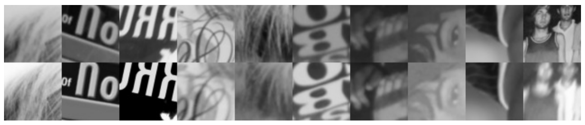


Figure 1. Example matching pairs from the *CDVS* patch dataset.

proposed by Winder and Brown [5], and evaluate the different compression schemes on these patch-level data sets. The higher the patch-level performance of a low bitrate descriptor, the higher its performance for pairwise matching and retrieval. The relationship between feature-level ROC performance and system level pairwise matching and retrieval performance is explored in [6]. In this work, we propose a scheme based on Entropy Constrained Vector Quantization and greedy rate allocation, which we expect to be close to the performance bound that can be achieved by any compression scheme. We compare feature-level ROC performance of different compression schemes at low bitrates of interest (30-120 bits).

The outline of the paper is as follows. In Section 2, we discuss the evaluation framework used. In Section 3, we propose a greedy algorithm for allocating bits across the different spatial bins of a Histogram-of-Gradients (HoG) descriptor. Finally, in Section 4, we compare the different compression schemes proposed in the course of the MPEG-CDVS standard.

2. Evaluation Framework

We create a patch dataset *CDVS* with 100,000 matching pairs [7], using the same methodology as [8]. The *CDVS* datasets consists of canonical scaled and oriented patches around Difference-of-Gaussian interest points from the 5 data sets used in the MPEG-CDVS standard: *Graphics*, *Paintings*, *Video*, *Buildings* and *Common Objects* [2]. The Winder and Brown [5], [8] data sets consist of patches from a small number of objects: *State of Liberty*, *Notredame* and *Trevis Fountain*, while the proposed *CDVS* data sets are drawn from images of thousands of objects and tens of thousands of matching image pairs. For creating matching pairs of patches, we perform image-matching with SIFT features and affine RANSAC to obtain feature correspondences and their corresponding canonical patches, using matching image pairs in the CDVS data sets [2]. Each canonical patch is 64×64 pixels. Non-matching pairs are obtained by considering pairs of patches that do not contain the same object. Example matching pairs from the *CDVS* dataset are shown in Figure 1. To incorporate interest point jitter statistics, patches are warped using small random similarity warps with position, rotation and scale noise. The standard deviation of synthetic noise added to *CDVS* datasets is 0.4 pixels, 11 degrees and $2^{0.12}$ octaves in orientation, shift and scale respectively, as recommended by Winder and Brown in [8]. The *CDVS* patch data sets are made available at [7].

Low FPR points, in the range of 10^{-4} to 10^{-1} are of interest on the ROC curve, as shown in [6]. For comparing different compression schemes, we use 25,000 matching pairs and $10\times$ as many non-matching pairs of patches (250,000). For training parameters for the rate allocation scheme, we use a separate set of 25,000 patches from the *CDVS* data set. For compression algorithms that require training codebooks, we use the 25,000 images from the MIRFLICKR [9] dataset.

3. Greedy Rate Allocation

The greedy rate allocation scheme proposed in this section can be applied to any compression scheme: here, we show how it can be applied to A_{m-1} lattice coding and ECVQ schemes proposed in [1]. A_{m-1} lattice coding and ECVQ schemes are first discussed in Section 3.1. We modify the SIFT descriptor slightly - we ignore the magnitude weighting in the descriptor computation step, and the global normalization step at the end [10]. Instead, we L_1 normalize the angular distribution of gradients in each spatial bin. This modified descriptor is called the *SIFT-Angle* descriptor, and the representation is then quantized using A_{m-1} lattice quantization or ECVQ.

3.1. Quantization of distributions

3.1.1. A_{m-1} **Lattice Quantization..** Let m represent the number of gradient bins (8 for *SIFT-Angle* descriptor). Let $P = [p_1, p_2, ...p_m] \in R^m_+$ be the original distribution as described by the gradient histogram, and $Q = [q_1, q_2, ...q_m] \in R^m_+$ be the quantized probability distribution defined over the same sample space. For A_{m-1} lattice coding, given a parameter κ, a A_{m-1} lattice of distributions $Q = Q(\gamma_1, \dots, \gamma_m)$ is constructed with probabilities

$$q_i = \frac{\gamma_i}{\kappa}, \quad \gamma_i, \kappa \in Z_+, \quad \sum_i \gamma_i = \kappa \tag{1}$$

The paramater κ controls the fidelity of quantization. The higher the value of κ parameter, higher the fidelity. The total number of lattice points $K(m, \kappa)$ is the number of partitions of κ into m terms $\gamma_1 + \ldots + \gamma_m = \kappa$ given by $K(m, \kappa) = \binom{\kappa+m-1}{m-1}$, implying that the rate $R(m, \kappa)$ needed for encoding is upper-bounded according to $R(m, \kappa) \leq \log_2 K(m, \kappa) \sim (m-1)\log_2 \kappa$. The quantized distribution is mapped to an index [1], and further compressed with arithmetic coding.

3.1.2. Entropy Constrained Vector Quantization.. For ECVQ, we use the Generalized Lloyd algorithm for quantizing the probability distributions [11], with KL divergence as the distance measure. Separate centroids are trained for each spatial bin of the descriptor. The vector quantized distribution is mapped to an index and transmitted with entropy codes.

More details of both quantization schemes can be found in [1]. To achieve a target bitrate budget for each descriptor, we need to allocate bits across the different spatial bins of the descriptor. There are an exponential number of combinations possible when assigning rate to different spatial bins of the SIFT descriptor. E.g., if there are k possible values for quantization parameter $\kappa_i \in [1, k]$, and n spatial bins, there are a total of k^n possible rate allocation schemes. Even for small values of k and n, this number is large and it is not feasible to consider all possible combinations. Hence, we need an efficient algorithm for finding allocations that lie close to or on the convex hull of the optimal rate-error trade-off curve. Here, we propose a greedy algorithm for allocating bits to different spatial bins of a descriptor given an overall bit budget.

3.2. Rate Allocation Algorithm

The rate allocation problem here is similar to the rate allocation problems that arise in the context of subband coding, and transform coding in image and video compression [12]. Let n be the number of spatial bins in the descriptor. Let $\kappa_1, \kappa_2 \ldots \kappa_n$ be the quantization parameters for each spatial bin. Let $\bar{\kappa} = (\kappa_1, \kappa_2 \ldots \kappa_n)$ be the vector allocation representing quantization parameters. Let $r_1, r_2 \ldots r_n$ or $r_1(\kappa_1), r_2(\kappa_2) \ldots r_n(\kappa_n)$ be the rates allocated to each spatial bin, and the rate for a given vector $\bar{\kappa}$ is given by $r(\bar{\kappa}) = \sum_{i=1}^{n} r_i(\kappa_i)$. Let $\bar{r} = (r_1, r_2 \ldots r_n)$ be the vector allocation of rates for the spatial bins. Let $\Gamma(\bar{\kappa})$ be the error rate for quantization parameters $\bar{\kappa}$. Let Q_{min} and Q_{max} be the minimum and maximum quantization parameters for each spatial bin.

The optimization problem for an overall rate constraint R, and error rate $\Gamma(\bar{\kappa})$ can be stated as follows

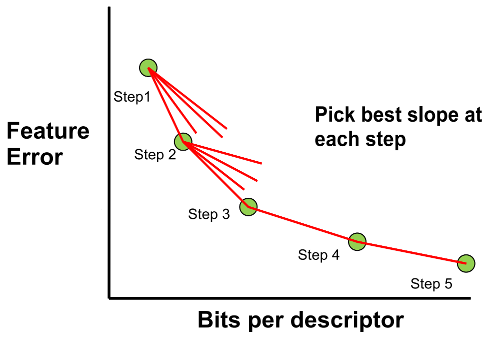
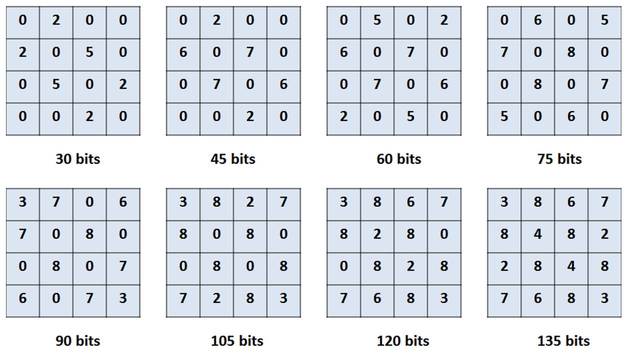


Figure 2. Illustration of downward pass of the greedy rate allocation algorithm. At each step in the algorithm, we pick the step that gives the best bitrate-error rate tradeoff (slope).



0	2	0	0
2	0	5	0
0	5	0	2
0	0	2	0

30 bits

0	2	0	0
6	0	7	0
0	7	0	6
0	0	2	0

45 bits

0	5	0	2
6	0	7	0
0	7	0	6
2	0	5	0

60 bits

0	6	0	5
7	0	8	0
0	8	0	7
5	0	6	0

75 bits

3	7	0	6
7	0	8	0
0	8	0	7
6	0	7	3

90 bits

3	8	2	7
8	0	8	0
0	8	0	8
7	2	8	3

105 bits

3	8	6	7
8	2	8	0
0	8	2	8
7	6	8	3

120 bits

3	8	6	7
8	4	8	2
2	8	4	8
7	6	8	3

135 bits

Figure 3. κ parameters for rate allocation scheme at different bitrates. Higher κ refers to higher bitrate.

$$\begin{aligned} \underset{\bar{\kappa}}{\text{minimize}} \quad & \Gamma(\bar{\kappa}) \\ \text{subject to} \quad & \sum_{i=1}^{n} r_i(\kappa_i) \leq R \end{aligned}$$

We explain our choice of functional for the error rate Γ later in this section. The error rate Γ decreases as the overall rate R increases. We propose a greedy algorithm which starts off with a minimum rate for each spatial bin, and adds rate incrementally to the descriptor in a greedy fashion. We call this the downwards pass, as rate increases and error decreases in each step of the algorithm, from start to finish. Alternately, the greedy algorithm can also start with a maximum rate for each spatial bin, and rate is removed in a greedy fashion from each spatial bin.

Let $\bar{\kappa}_0, \bar{\kappa}_1 \dots \bar{\kappa}_r$ be the quantization parameters in iteration number $0, 1, \dots r$ respectively. Here, we describe the algorithm for the downward pass. The proposed algorithm is similar to iterative algorithms that have been proposed for the subband coding problem [12].

- Determine the initial bit allocation $\bar{\kappa}_0$ by assigning to each spatial bin the smallest possible bit rate.
- Let vectors $\bar{\kappa_k}$ and $\kappa_{\bar{k+1}}$ be the quantization parameter for current and next iterations respectively. Define a set of vectors $\bar{\delta}_i, \quad \forall i \in [1, n], \quad \forall j \in [1, n]$

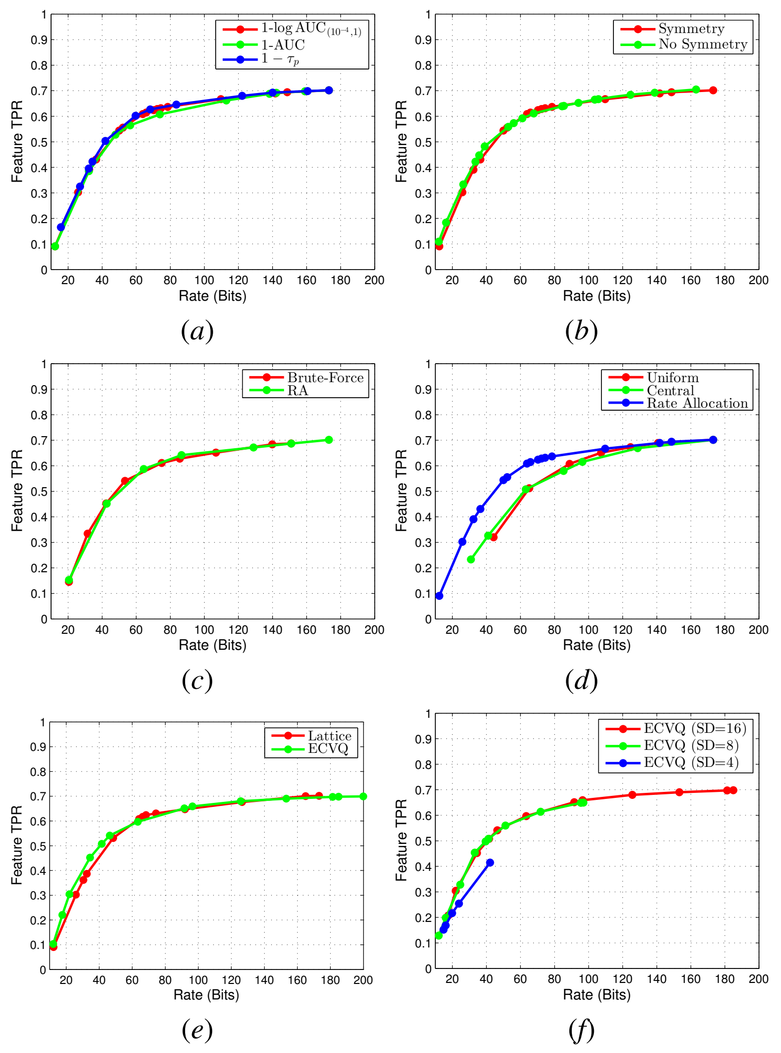


Figure 4. Rate allocation experiments. We compare TPR at low FPR point of 0.001. Similar results are observed for other FPR points. In Figure (*a*), we note that the different error measures perform comparably. In Figure (*b*), we note that imposing spatial symmetry constraints in rate allocation performs comparably to when no spatial symmetry is imposed. In Figure (*c*), we observe that the greedy rate allocation algorithm performs on par with brute force for step size $\epsilon = 8$. In Figure (*d*), we observe that the rate allocation scheme outperforms naive schemes. In Figure (*e*), we note that lattice coding with rate allocation comes close to the performance of ECVQ with rate allocation. In Figure (*f*), performance does not improve by considering larger sub-divisions of the descriptor beyond individual spatial bins.

$$\bar{\delta}_i[j] = \begin{cases} \epsilon & \text{if } j = i \\ 0 & \text{if } j \neq i \end{cases} \tag{2}$$

where ϵ is the step size in the algorithm. Larger the ϵ, the fewer the steps in the algorithm. For $i \in 1 \dots n$, we define the set of vectors $\bar{t_i}, \quad \forall i \in [1, n], \quad \forall j \in [1, n]$,

$$\bar{t_i}[j] = \bar{\kappa_k}[j] + \bar{\delta_i}[j] \tag{3}$$

Next, we compute the set of slopes:

$$s(\bar{t_i}, \bar{\kappa_k}) = \frac{r(\bar{t_i}) - r(\bar{\kappa_k})}{\Gamma(\bar{t_i}) - \Gamma(\bar{\kappa_k})} \tag{4}$$

The parameter $\bar{\kappa_{k+1}}$ can be determined as:

$$\bar{\kappa_{k+1}} = \underset{\bar{t_i}}{\operatorname{argmin}}\ s(\bar{t_i}, \bar{\kappa_k}), i \in [1, n] \tag{5}$$

- Store the next rate $r(\bar{\kappa_{k+1}})$ and the new error rate $\Gamma(\bar{\kappa_{k+1}})$ for $\bar{\kappa_{k+1}}$ as computed previously in Step (4). If the desired rate is sufficiently close to target bitrate R, stop the algorithm.
- Repeat (2)-(3) till (3) satisfies the target bitrate criterion.

We note that the number of iterations r is given by $r = n\lceil \frac{Q_{max}}{\epsilon} \rceil$. As ϵ increases, there are fewer iterations, but we have coarser resolution in rate.

3.3. Experimental Results

We consider A_{m-1} lattice quantization scheme described in Section 3.1, and $\mathcal{S} = 16$ spatial bins of the SIFT-Angle descriptor to demonstrate the rate allocation algorithm. For A_{m-1} lattice coding, the rate is varied using lattice quantization parameter κ_i. We vary κ_i from 2 to a maximum of 8, resulting in 36 to 6435 lattice points. For each scheme, we plot the convex hull over the set of achievable rate and feature-level TPR at fixed low FPR (10^{-3}). Similar results are obtained for the rate allocation experiments at other low FPR points (10^{-2},10^{-1}). Step size parameter $\epsilon = 1$, with miminum $\kappa_i = 2$.

3.3.1. Comparing different error measures Γ. We explore different error measures for the feature error in (4). We apply different error measures to maximize TPR in the low FPR range. In Figure 4 (*a*), we apply three error measures: (a) $\Gamma = 1 - \log \text{AUC}_{(10^{-4},1)}$, where $\log \text{AUC}_{(10^{-4},1)}$ is the area under the ROC curve in the log-domain in the FPR range of 10^{-4} to 10^{-1}, (b) $\Gamma = 1 - \text{AUC}$, where AUC is the area over the entire curve in the linear domain, and (c) $\Gamma = 1 - \tau_p$ for a specific FPR point 10^{-3} on the ROC curve. We plot the TPR at low FPR point 10^{-3} for the different error measures. We note that $1 - \log \text{AUC}_{(10^{-4},1)}$ performs slightly better than $1 - \text{AUC}$ at the low FPR points, as the $\log \text{AUC}_{(10^{-4},1)}$ measure emphasizes low FPR points on the ROC curve. We note that $1 - \tau_p$ performs the best for FPR=10^{-3}. The error measure, $1 - \tau_p$, can be used to optimize performance for a specific point on the ROC curve. $1 - \log \text{AUC}_{(10^{-4},1)}$ performs well across the range of low FPR and hence is the error functional of choice for further experiments.

3.3.2. Lattice parameters with spatial symmetry. We impose spatial symmetry in the rate allocation scheme, i.e., the same number of bits are allocated (by applying same δ) to spatial bins that are symmetric about the center of the patch. The symmetry pattern is shown in the quantization parameter matrices shown in Figure 3. In Figure 4 (*b*), we note that imposing spatial symmetry about the origin for rate allocation performs on par when no symmetry condition is imposed. We show κ parameters that are obtained from the rate allocation scheme with the symmetry constraint in Figure 3. We note that more bits (as reflected by higher κ) are allocated to the central spatial bins compared to the outer spatial bins. This is intuitive as patch distortions are higher in the peripheral spatial bins compared to the central ones. Also, we observe that adjacent spatial bins are not assigned a high number of bits at low rates. Adjacent spatial bins have similar gradient statistics and allocating bits to non-adjacent bins improves performance.

3.3.3. Comparing rate allocation to brute force. In Figure 4 (*c*), we compare the greedy rate allocation scheme (*RA*) to the brute force scheme (*Brute-Force*). Step size for the rate allocation algorithm ($\epsilon = 8$) in (2). $\epsilon = 8$ selects $\kappa = 0$ or $\kappa = 8$ for each spatial bin. For brute force, 65536 possibilities are considered (2^{16}), with $\kappa = 0, 8$ for each spatial bin. We note that the greedy allocation scheme performs on par with the brute-force scheme. With smaller step sizes, the brute force scheme becomes quickly intractable.

3.3.4. Comparing rate allocation to basic schemes. We compare the greedy rate allocation scheme to two naive schemes: (1) *Uniform* scheme assigns the same quantization parameter κ to all spatial bins by increasing the quantization parameter in steps of 1 for all spatial bins. (2) *Central* scheme allocates maximum rate (quantization parameter $\kappa = 8$) to spatial bins starting from the patch center and moving outwards. We observe that the proposed rate allocation scheme significantly outperforms both naive schemes at low bitrates, as shown in Figure 4 (*d*).

3.3.5. Comparing lattice coding and ECVQ. For the ECVQ rate allocation scheme, ECVQ codebooks are trained for each spatial bin of the SIFT-Angle descriptor, and rate is incrementally added using the greedy algorithm. KL divergence is used as the distance measure. The maximum number of centroids is set to 8192, corresponding to a maximum of 13 bits for each spatial bin. The minimum rate for each spatial bin is set to 4 bits. We add rate incrementally to each spatial bin with step size of 2 bits. Codebooks with corresponding rate parameters are chosen for each spatial bin at different bit rates. In Figure 4 (*e*), we observe that the lattice coding scheme comes close to the performance of ECVQ after rate allocation at the different FPR points. We expect the proposed ECVQ coding with greedy rate allocation to be close to the upper bound on performance any scheme could achieve.

We evaluate the performance of the ECVQ rate allocation scheme for different sub-divisions of the SIFT-Angle descriptor. The number of sub-divisions $SD = 16, 8, 4$ corresponds to grouping $SB = 1, 2, 4$ adjacent spatial bins of the 16 spatial bins of the SIFT-Angle descriptor. We use a maximum codebook size of 8192 centroids for each SD. In Figure 4 (*f*), we compare the performance for varying SB and note that performance does not improve for $SB > 1$.

4. Comparisons of Different Schemes

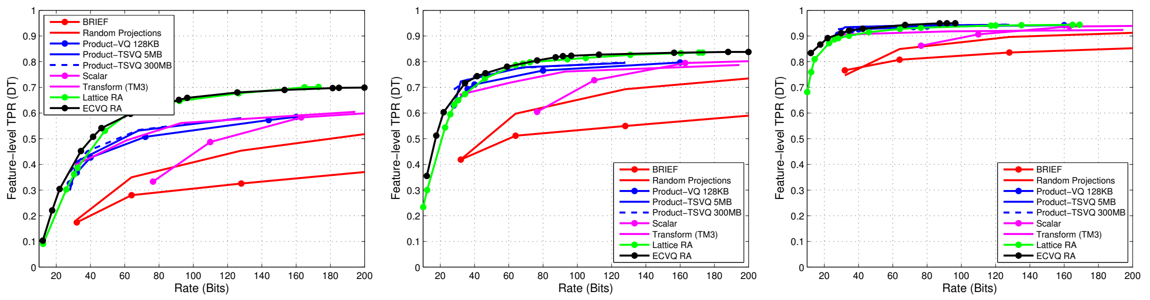
In Figures 5,6,7, we compare the following schemes:

- (*BRIEF*): Binarized intensity differences between random pixel location pairs as proposed in the BRIEF descriptor [13].
- (*Random Projections*): Random projections based on Locality Sensitive Hashing [14]
- (*Scalar*): Scalar quantization of each dimension followed by entropy coding with an arithmetic coder [15]
- (*TM3*): Linear transform coding technique based on sums and differences of adjacent gradient bins [16]
- (*Product-VQ 128 KB*): Product Vector Quantization with 128 KB of tables [4]
- (*Product-TSVQ 5 MB, Product-TSVQ 300 MB*): Product Tree Structured Vector Quantization with 30 MB of tables [17], and 300 MB of tables [4]
- (*Lattice RA*): Lattice coding with greedy rate allocation, $SB = 16$ (see Figure 4)
- (*ECVQ RA*): Entropy Constrained Vector Quantization with greedy rate allocation, $SB = 16$, (see Figure 4)

For a more detailed description of the schemes, readers are referred to [6].

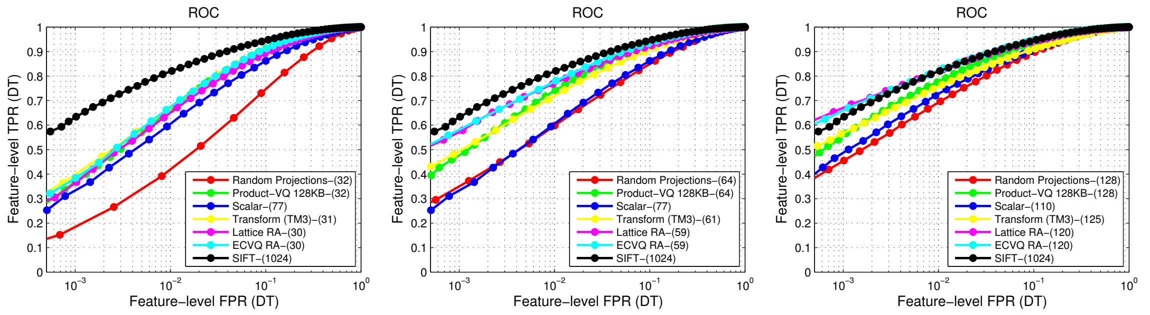
For Distance Threshold matching (DT), we plot the TPR at typical low FPR points $(10^{-3}, 10^{-2}, 10^{-1})$ in Figure 5. One can generate similar ROC curves for Nearest Neighbor matching based on the Ratio Test (NNRT) [6], [10]. For NNRT, we measure how often matching and non-matching pairs satisfy the ratio-test criterion, in the presence of a fixed number of background features. A different set of background features N_b are drawn randomly for each descriptor, from the same dataset. A smaller ratio test threshold will result in a lower TPR and lower FPR. For Nearest Neighbor Matching with Ratio Test (NNRT), we set the number of background features to 400, and vary the ratio test threshold between 0.65 and 0.95 to sweep the feature-level ROC. Similar results are obtained when the number of background features is varied (see [6] for more results). Full ROCs of low bitrate descriptors at 30, 60 and 120 bits are presented in Figures 6 and 7 for DT and NNRT respectively. We use KL divergence as the distance measure for all distribution based descriptors, and L_2 distance measure for SIFT. We make the following salient observations:

- In Figure 5 (DT), we note that there is a significant gap in performance between BRIEF/Random Projections and other schemes.
- In Figure 5 (DT), we note that the performance for *Product VQ* and *Product TSVQ* improves as the size of the codebook increases, but only marginally. The gap between the scheme that requires 128 KB and 380 MB is small. These feature-level results are consistent with image level pairwise matching results reported in [18].
- In Figure 5 (DT), we note that ECVQ with rate allocation (*ECVQ RA*) performs the best for all data sets. We expect this scheme to be close to the bound in performance any compression scheme can achieve. The lattice coding scheme with rate allocation (*Lattice RA*), Transform coding (*Transform (TM3)*) and Product-TSVQ schemes come close to the performance of (*ECVQ RA*). At high FPR point 10^{-1}, we note that the *Product TSVQ* scheme performs slightly better than *ECVQ RA*. This is because the codebook size for *ECVQ RA* is set to 8192 centroids (1 MB), compared to the 380 MB for the *Product TSVQ* scheme.
- In Figure 5 (DT), we note the ranking in performance of different schemes changes at different FPR points. E.g., *Lattice RA* performs better than *Product VQ* at low FPR point(10^{-3}), but performs worse at higher FPR point (10^{-1}). A similar observation is made when comparing *Transform TM3* and *Product VQ*. We note that close to 100 bits are required to match the performance of SIFT for the *CDVS* dataset.
- In Figure 7 (NNRT), we compare the subset of schemes: *Product VQ (128 KB)*, *Transform (TM3)*, *Lattice RA*, *BRIEF* and *Random Projections*, for NNRT matching. In Figure 7, we note that *Product VQ (128 KB)* and *Transform (TM3)* coding outperform *Lattice RA* at 30 bits for the *CDVS* data set by a small margin. At 60 bits, the *Lattice RA* scheme outperforms the rest of the schemes. We note that at 60 bits, there is still a significant gap in performance between 1024-bit uncompressed SIFT and *Lattice RA*. The performance gap between uncompressed SIFT and *Lattice RA* at 128 bits is small.
- Comparing performance in Figures 6 and 7, it might seem that DT matching performs better than NNRT at the feature level. However, note that DT produces far higher $O(N_b^2)$ false positive feature matches as every feature is compared to every other feature using a distance threshold. On the other hand, NNRT produces $O(N_b)$ false positive feature matches. As a result, at the image level, for DT, either the inlier ratio is too small for RANSAC to find the correct model (lowering image-level TPR), or invalid models are found by RANSAC due to the many false-positive feature matches (increasing image-level FPR), leading to worse image-level FPR, TPR performance compared to NNRT. Readers are referred to Chapter 3 [6] for a detailed comparison of image-level TPR and FPR for DT and NRRT matching.



CDVS (FPR=10^{-3}) (a) *CDVS (FPR=10^{-2})* (b) *CDVS (FPR=10^{-1})* (c)

Figure 5. TPR comparisons between different schemes at FPR points $10^{-3}, 10^{-2}, 10^{-1}$ for Distance Threshold matching. We expect *ECVQ RA* to be close the bound in performance. The ranking of different schemes depends on FPR operating point.



CDVS (30) (a) *CDVS (60)* (b) *CDVS (120)* (c)

Figure 6. Full ROC comparisons between different schemes for 30, 60 and 120 bit descriptors for Distance Threshold matching.

5. Conclusion

We provide detailed patch-level ROC comparisons of different compression schemes proposed during the MPEG-CDVS standard. We propose a greedy rate allocation scheme for assigning bits to different spatial bins of the SIFT-descriptor. We present results for ECVQ combined with the greedy rate allocation scheme, which we expect to be close the performance bound that can be achieved by any compression scheme. For patch-level experiments, we note that at 30 bits per descriptor, Transform Coding [16] and Product VQ [18] outperform A_{m-1} Lattice Coding with greedy rate allocation. At 60 and 120 bits per descriptor, A_{m-1} Lattice Coding scheme with greedy rate allocation outperforms Product VQ and Transform Coding.

References

[1] V. Chandrasekhar, G. Takacs, D. M. Chen, S. S. Tsai, R. Grzeszczuk, Y. Reznik, and B. Girod, "Compressed Histogram of Gradients: A Low Bitrate Descriptor," *International Journal of Computer Vision, Special Issue on Mobile Vision*, vol. 96, no. 3, pp. 384–399, January 2012.

[2] "Compact Descriptors for Visual Search: Evaluation Framework," *ISO/IEC JTC1 SC29 WG11 output document N12202*, July 2011.

[3] "Compact Descriptors for Visual Search: Call for Proposals," *ISO/IEC JTC1 SC29 WG11 output document N12201*, July 2011.

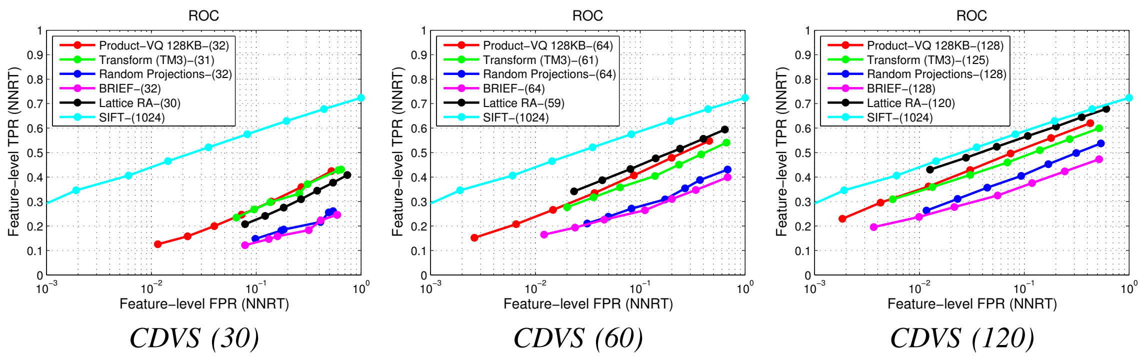


Figure 7. Full ROC comparisons between different schemes for 30, 60 and 120 bit descriptors for Nearest Neighbor matching with a Ratio Test. The closest available bitrate to the target bitrates for the different schemes are listed in each figure. At 30 bits, *Transform (TM3)* and *Product VQ (128 KB)* outperform *Lattice RA*. *Lattice RA* outperforms all other low bitrate schemes at 60 and 120 bits.

[4] "Test Model 2: Compact Descriptors for Visual Search," *ISO/IEC JTC1 SC29 WG11 output document W12734*, April 2012.

[5] S. Winder, G. Hua, and M. Brown, "Picking the Best DAISY," in *Proceedings of Computer Vision and Pattern Recognition (CVPR)*, Miami, Florida, June 2009, pp. 178–185.

[6] V. Chandrasekhar, "Low Bitrate Image Retrieval with Compressed Histogram of Gradient Descriptors," *Ph.D. thesis, Department of Electrical Engineering, Stanford University*, April 2013, http://blackhole1.stanford.edu/vijayc/vijay-thesis.pdf.

[7] *CDVS Patches*, 2013, http://blackhole1.stanford.edu/vijayc/cdvs_patches.tar.

[8] S. Winder and M. Brown, "Learning Local Image Descriptors," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Minneapolis, Minnesota, June 2007, pp. 1–8.

[9] B. T. Mark J. Huiskes and M. S. Lew, "New Trends and Ideas in Visual Concept Detection: The MIR Flickr Retrieval Evaluation Initiative," in *MIR '10: Proceedings of the 2010 ACM International Conference on Multimedia Information Retrieval*, New York, NY, USA, 2010, pp. 527–536.

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

[11] P. A. Chou, T. Lookabaugh, and R.M.Gray, "Entropy Constrained Vector Quantization," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 37, no. 1, pp. 31–42, Jan 1989.

[12] P.H.Westerink, J. Biemond, and D. E. Boekee, "An Optimal Bit Allocation Algorithm for Subband Coding," in *Proceedings of International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, New York, USA, April 1988, pp. 757–760.

[13] M. Calonder, V. Lepetit, and P. Fua, "BRIEF: Binary Robust Independent Elementary Features," in *Proceedings of European Conference on Computer Vision (ECCV)*, Crete, Greece, October 2010.

[14] C. Yeo, P. Ahammad, and K. Ramchandran, "Rate-efficient Visual Correspondences using Random Projections," in *Proceedings of IEEE International Conference on Image Processing (ICIP)*, San Diego, California, October 2008, pp. 217–220.

[15] V. Chandrasekhar, G. Takacs, D. M. Chen, S. S. Tsai, and B. Girod, "Transform Coding of Image Feature Descriptors," in *Proceedings of Visual Communications and Image Processing Conference (VCIP)*, San Jose, California, January 2009.

[16] S. Paschalakis, K. Wnukowicz, M. Bober, A. Mosca, M. Mattelliano, G. Francini, S. Lepsoy, and M. Balestri, "CDVS CE2: Local Descriptor Compression Proposal," *ISO/IEC JTC1 SC29 WG11 input document M25929*, July 2012.

[17] G. Francini, S. Lepsoy, M. Balestri, and G. Cordara, "Telecom Italia's response to the MPEG CfP for Compact Descriptors for Visual Search," *ISO/IEC JTC1 SC29 WG11 input document M22672*, November 2011.

[18] G. Francini, S. Lepsoy, and M. Balestri, "Telecom Italia Response to the CDVS Core Experiment 2," *ISO/IEC JTC1 SC29 WG11 input document M24737*, April 2012.