Low Latency Image Retrieval with Progressive Transmission of CHoG Descriptors

Vijay Chandrasekhar
Stanford University, CA
vijayc@stanford.edu

Sam S. Tsai
Stanford University, CA
sstsai@stanford.edu

Gabriel Takacs
Stanford University, CA
gtakacs@stanford.edu

David M. Chen
Stanford University, CA
dmchen@stanford.edu

Ngai-Man Cheung
Stanford University, CA
nmcheung@stanford.edu

Ramakrishna Vedantham
Nokia Research Center, CA
ramakrishna.vedantham@nokia.com

Yuriy Reznik
Qualcomm Inc., CA
yreznik@qualcomm.com

Radek Grzeszczuk
Nokia Research Center, CA
radek.grzeszczuk@nokia.com

Bernd Girod
Stanford University, CA
bgirod@stanford.edu

ABSTRACT

To reduce network latency for mobile visual search, we propose schemes for progressive transmission of Compressed Histogram of Gradients (CHoG) descriptors. Progressive transmission reduces the amount of transmitted data and enables early termination on the server, thus reducing end-to-end system latency. With progressive transmission of CHoG descriptors, we are able to reduce network latency to ∼1 second in a 3G network. We report a 4× decrease in end-to-end system latency compared to transmitting uncompressed SIFT descriptors or JPEG images.

Categories and Subject Descriptors

C.5.0 [**Computer Systems Organization**]: Computer Systems Implementation—*General*

General Terms

Algorithms,Design

Keywords

mobile visual search, CHoG, content-based image retrieval

1. INTRODUCTION

Mobile phones have evolved into powerful image and video processing devices, equipped with high-resolution camera, color displays, and hardware-accelerated graphics. They are also equipped with location sensors, GPS receivers, and connected to broadband wireless networks allowing fast transmission of information. This enables a class of applications which use the camera phone to initiate search queries about objects in visual proximity to the user. Such applications can be used for identifying products, comparison shopping, finding information about movies, CDs, real estate or products of the visual arts. Google Goggles [1] and Nokia Point and Find [2] are examples of recently developed commercial applications. For these applications, a query photo is taken by a mobile device and compared against database photos on a remote server. A set of image feature descriptors is used to assess the similarity between the query photo and each database photo. In designing such systems, it is important to ensure fast and accurate retrieval of the results.

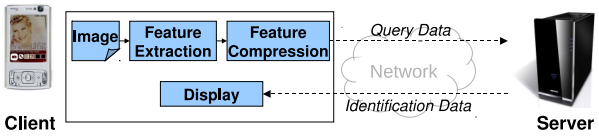


Figure 1: A mobile CD cover recognition system where the server is located at a remote location. Feature descriptors are extracted on the mobile-phone and query feature data is sent over the network. Once the CD cover is recognized on the server, identification data is sent back to the mobile-phone.

The system latency can be broken down into 3 components: (a) Processing time on mobile client (b) Network transmission latency and (c) Processing time on server. In [16, 15], we show that processing on the server and client take approximately ∼1 second each, while the network transmission typically is the bottleneck in a 3G system. Hence, the size of the data sent over the network needs to be as small as possible to reduce latency and improve user interaction. To reduce network latency, we extract feature descriptors on the phone, compress the descriptors and transmit them over the network as illustrated in Figure 1. In this work, we focus on how system latency can be minimized using progressive transmission of query data.

1.1 Prior Work

In [15], we present a state-of-the-art mobile product recognition system using a camera phone. The product is recognized through an image-based retrieval system located on a

MCMC'10, October 29, 2010, Firenze, Italy.

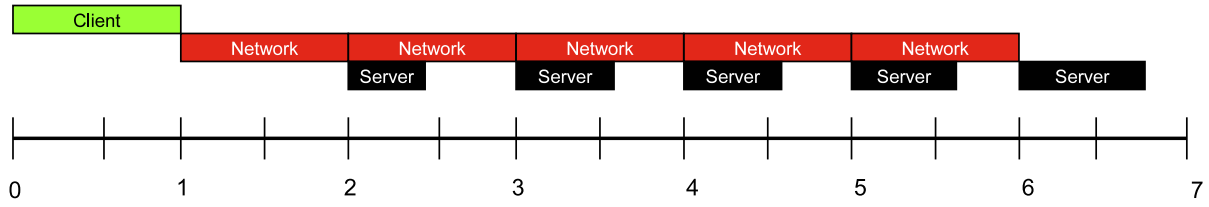


Figure 2: Typical timeline (seconds) for progressive transmission of feature data. Feature data are grouped and transmitted over the network. If the server finds a match with partial feature data, a response is sent back to the mobile client and data transmission is early terminated.

remote server. We provide experimental timings for different parts of the system, and show that transmission delay remains the bottleneck for 3G networks.

Low bitrate descriptors are critical for achieving low latency. Several different descriptors have been proposed in the literature, e.g., Scale Invariant Feature Transform (SIFT) [10], Speeded Up Robust Features (SURF) [3] and Gradient Location and Orientation Histogram (GLOH) [11]. In [6, 5], we propose a framework for computing low bitrate feature descriptors called Compressed Histogram of Gradients (CHoG). In [4], we perform a comprehensive survey and show that CHoG outperforms all SIFT compression schemes. We show that the CHoG descriptor at 60 bits matches the performance of the 128 dimensional 1024-bit SIFT descriptor [10].

Progressive transmission is common for images. E.g., JPEG2000 produces an embedded bitstream which allows for progressive transmission and rendering of images. This enables a client to display an image quickly by decoding only a portion of the image that it has received. As additional data are received, the image can be progressively improved. Here, we show how progressive transmission can be applied to feature descriptors. We focus on progressive transmission of CHoG descriptors, but the ideas are also applicable to SIFT [10], SURF [3] and GLOH [11].

1.2 Contributions

In this work, we discuss how to reduce network latency using progressive transmission of CHoG descriptors.

- We propose two progressive transmission schemes, one based on sequentially sending descriptors, and another based on spatial embedding of descriptor data. We discuss the protocol on client and server for scalable transmission of feature data.
- We show that progressive transmission of feature data reduces average transmission bitrate by $\sim$4$\times$ compared to transmitting full data.
- We analyze network latency in a 3G network for retrieval systems based on CHoG, SIFT and JPEG. With progressive transmission, we are able to reduce network latency to $\sim$1 second in a 3G network. We report a 4$\times$ decrease in end-to-end system latency compared to uncompressed SIFT descriptors or JPEG images.

1.3 Outline

We organize the paper as follows. In Section 2, we review the CHoG descriptor and discuss how CHoG descriptors can be transmitted in a rate-scalable manner. In Section 3, we provide an overview of the retrieval system. In Section 4, we provide experimental results for our retrieval system.

2. COMPRESSED HISTOGRAM OF GRADIENTS

In Section 2.1, we briefly review the CHoG descriptor, and in Section 2.2, we discuss techniques for progressive transmission on the client once descriptors are computed.

2.1 Descriptor Review

CHoG [6] is a Histogram of Gradients descriptor that is designed to work well at low bitrates. We highlight some key aspects of the descriptor here and readers are referred to [6, 5] for more details.

First, the patch is divided into soft log polar spatial bins using DAISY configurations proposed in [18]. The DAISY-9 configuration is illustrated in Figure 3. The descriptor directly encodes the joint (d_x, d_y) gradient histogram in each spatial bin. This allows the use of distance measures such as Kullback Leibler (KL) divergence, and enables efficient quantization and compression. Typically, 9 to 13 spatial bins and 3 to 9 gradient bins are chosen resulting in 27 to 117 dimensional descriptors.

We quantize the gradient histogram in each cell individually and map it to an index. The indices are encoded with fixed length or entropy codes, and the bitstream is concatenated together to form the final descriptor. In prior work [6, 5], we have explored several schemes for histogram quantization and compression: Huffman Trees, Type Coding and optimal Lloyd-Max Vector Quantization. In this work, we use Type Coding, which is linear in complexity to the number of histogram bins and performs close to optimal Lloyd-Max VQ [5].

2.2 Progressive Transmission

Progressive transmission of features is illustrated in Fig. 2. Server processing and network transmission time are overlapped to reduce end-to-end system latency. Client processing and network transmission could be overlapped as well. However, this yields limited benefits, as 80% of the time on the mobile client is spent in interest point detection i.e., detecting the location, scale and orientation of features, which has to be carried out before computing feature descriptors [15].

In this section, we discuss two schemes for transmitting feature data progressively over the network. If a match is

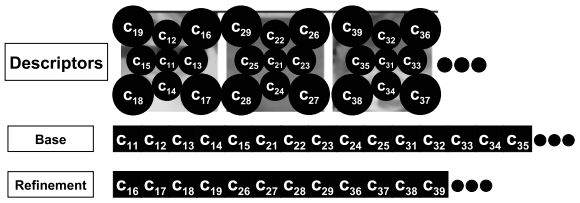


Figure 3: Transmission order of bits on the client when spatial embedding is employed. The DAISY-9 spatial bin configuration is overlaid on a set of scaled and oriented patches. c_{ij} refers to the encoded distribution in i^{th} feature and j^{th} spatial bin. The encoded distributions belonging to the inner spatial bins are transmitted first (*Base Layer*), followed by the outer spatial bins (*Enhancement Layer*).

found on the server with partially received data, a response is sent back to the mobile client.

2.2.1 Sequential Transmission of Features

One simple technique for progressive transmission is to sequentially transmit feature descriptors. Typically, as the number of descriptors increases, classification accuracy increases. Each keypoint has an associated Hessian blob response. Blobs with low Hessian responses often correspond to edge-like regions which are poorly localized. We sort descriptors by their Hessian responses, assemble the feature data in blocks of 100 descriptors, and transmit them over the network. If a match is found on the server with partially received feature data, a response is sent back to the client signalling for early termination of data transmission.

2.2.2 Spatial Embedding of Features

Another progressive transmission technique involves the order in which components of each feature descriptor are sent over the network. Each spatial bin in the DAISY configuration is encoded independently from others in the CHoG descriptor. The DAISY spatial binning naturally lends itself to progressive transmission. For a set of descriptors, we first transmit the encoded data belonging to inner spatial bins in the DAISY configuration for all descriptors, followed by data for the outer spatial bins. Descriptor data, which uses only the inner spatial bins are less discriminative, and hence, result in lower classification accuracy. Once full descriptor data are received, classification accuracy can be improved. This spatial embedding technique is illustrated in Figure 3. We restrict the number of rings to two in the DAISY configuration as we do not observe any improvement in performance beyond two rings. In the following, we will use data from inner spatial bins as the *Base Layer*, and data from outer spatial bins as *Enhancement Layer*. We first discuss the retrieval pipeline before presenting results for the progressive transmission schemes proposed in this Section.

3. RETRIEVAL SYSTEM

In this section, we provide an overview of the processing on the client and the server.

3.1 Client

We extract CHoG descriptors on the mobile device and transmit them over the network as illustrated in Figure 1.

Layer	Bits / descriptor
Base	37
Base+Enhancement	70

Table 1: CHoG bitrates for DAISY-9 spatial binning and VQ-7 gradient binning. *Base Layer* refers to the inner 5 spatial bins. *Enhancement Layer* refers to the outer 4 spatial bins. The 70-bit CHoG descriptor is chosen as it performs on par with the 1024-bit SIFT descriptor.

Figure 4: Example image pairs from the dataset. A clean database picture (*top*) is matched against a real-world picture (*bottom*) with various distortions.

Feature extraction can be carried out in less than 1 second on current generation smart phones making this approach feasible [14]. We choose 9 spatial bins, 7 gradient bins, and a ∼70 bit type-quantized CHoG descriptor that performs on par with the 1024-bit SIFT descriptor [5]. The DAISY-9 spatial binning configuration is illustrated in Figure 3. For DAISY-9 spatial binning, we use the inner 5 spatial bins as the *Base Layer* and the outer 4 spatial bins as the *Enhancement Layer*. The parameters for the chosen CHoG descriptor are shown in Table 1.

We extract 100 to 700 CHoG descriptors on the mobile device and transmit them over the network. The location data for features is compressed using the histogram compression scheme proposed in [17]. The location data is transmitted along with *Base Layer* descriptor data. We report experimental data for CHoG parameters chosen in Table 1, but the results generalize to other parameters too.

3.2 Server

The retrieval pipeline for CHoG descriptors builds on techniques proposed in [12, 13]. We use soft-assignment for quantization of descriptors to the 3 nearest centroids in each VT [13]. For each VT, we use the standard Term Frequency-Inverse Document Frequency (TF-IDF) scheme [12] that represents query and database images as sparse vectors of visual word occurrences, and compute a similarity between each query and database vector. We use the weighting scheme proposed in [12] which reduces the contribution of less descriminative descriptors. We use geometric constraints to re-rank the list of top 500 images [9]. Finally, we consider up to 50 images for pairwise matching with a RANSAC [8] affine consistency check.

4. RESULTS

In Section 4.1, we describe the data set and parameters

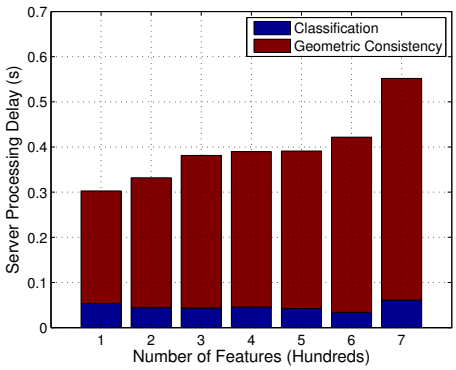


Figure 5: Server processing delay for CHoG descriptors when features are incrementally classified in the Vocabulary Tree. The time taken for geometric verification increaseas as the number of descriptors increases due to increased time spent in nearest neighbor computation in RANSAC.

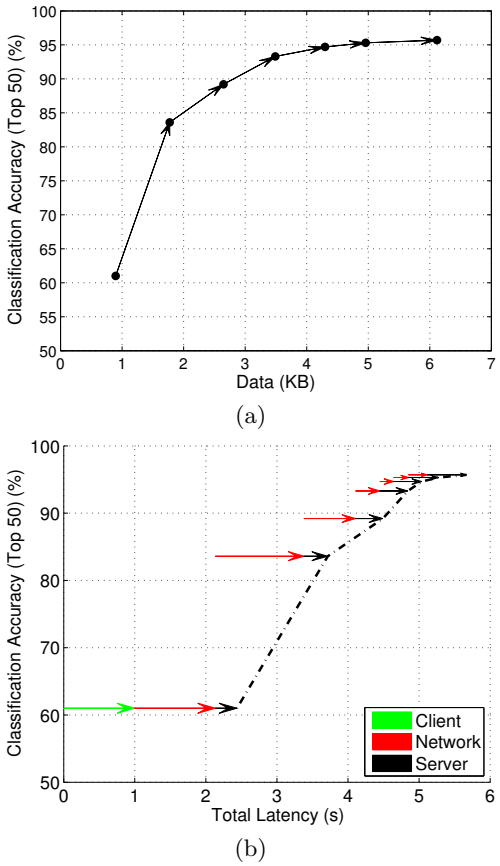


Figure 6: Figure (a) shows the accuracy profile for a typical CHoG descriptor. CA increases as the number of descriptors increases and eventually plateaus off. Figure (b) shows the timing profile at different CA. The server starts the matching process once feature data are received in blocks of 100. The dotted line in Figure (b) traces the CA profile in Figure (a). The overlap in network transmission and server processing enables effective early termination for a majority of query images.

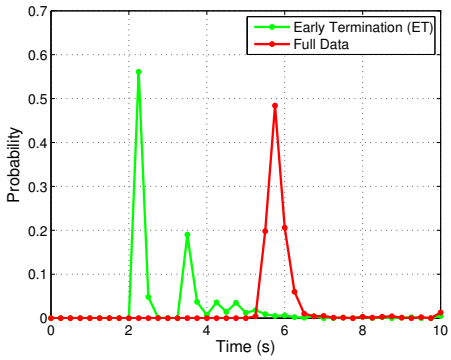


Figure 7: Distribution of end-to-end system latency with Early Termination (ET). We note that ET can be achieved for most query images and average latency can be reduced significantly compared to when full feature data are transmitted.

used in our experiments. Next, in Section 4.2, we discuss results for progressive transmission of CHoG descriptors using techniques proposed in Section 2.2.1. Finally, we compare latency of transmitting CHoG descriptors to JPEG compressed images and SIFT descriptors.

4.1 Experimental Setup

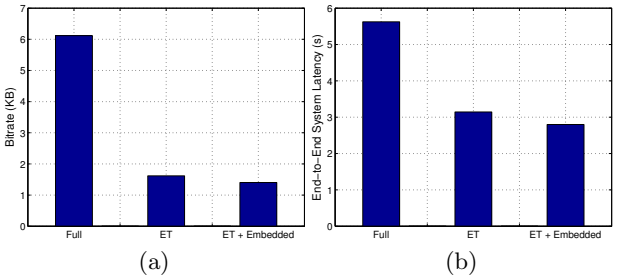
For evaluation, we use a database of one million CD, DVD and book cover images, and a set of 1000 query images [7] exhibiting challenging photometric and geometric distortions, as shown in Figure 4. Each image has 500 × 500 pixels resolution. We define Classification Accuracy (CA) as the percentage of query images correctly retrieved after RANSAC. We set the minimum number of matching features after RANSAC geometric verification to 12, which is high enough to avoid any false positive matches. We report CA and latency timings for progressive transmission of CHoG descriptors. The data transmission experiments are conducted in a AT&T 3G wireless network, averaged over several days, with a total of more than 5000 transmissions at indoor locations where a image-based retrieval system would be typically used.

4.2 Progressive Transmission

We discuss the results for progressive transmission by sequential transmission of features transmitted, and transmission of the base layer followed by the enhancement layer.

4.2.1 Sequential Transmission of Features

First, in Figure 5, we study server processing delay as descriptors are successively received in blocks of 100. The server processing delay can be broken down into two parts: Feature Classification and Geometric Consistency Check (GCC). The Feature Classification time remains roughly constant as feature data are incrementally classified, while GCC time increases as the number of descriptors increases. GCC is carried out pairwise for each database image present in the top 50 candidates obtained from Vocabulary Tree voting, till a valid match is found. The GCC step consists of finding a set of putative feature matches by computing the nearest neighbor for each query descriptor from the set of database image descriptors. As the number of query descriptors increases, the time taken for nearest neighbor compu-



(a) (b)

Figure 8: Average reduction in bitrate and system latency with progressive transmission of feature data at 96% CA. *Full* = Full feature data, *ET* = Early Termination with sequential feature transmission, *ET + Embedded* = Early Termination with sequential feature transmission and spatial embedding. With progressive transmission, we achieve 4× reduction in data, and 2× reduction in system latency.

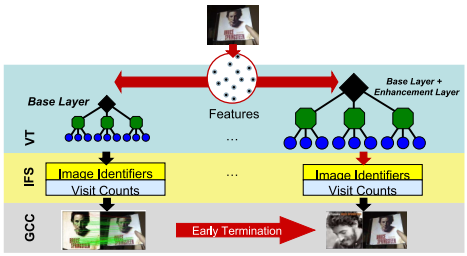


Figure 9: Server Architecture. We store multiple Vocabulary Trees on the server, one for each embedded layer. The server starts the recognition process once data for an embedded layer is received. If a match is found with *Base Layer* data, the server early terminates and returns a response to the mobile client.

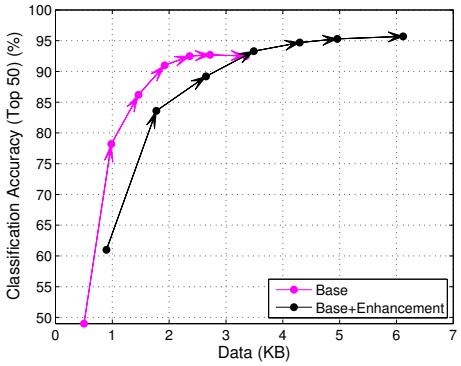


Figure 10: Accuracy profiles for *Base Layer* data and *Base + Enhancement Layer* data. Incrementally transmission of *Base Layer* data, followed by *Enhancement data* reduces system latency further.

tation increases, resulting in increased time spent in GCC. This trend is observed in Figure 5. We observe that incremental classification of features reduces processing time, and less than one second server latency can be achieved even after full feature data are received.

Next, in Figure 6 (a), we study the classification accuracys as the number of descriptors increases. The CA improves as the number of features increases and eventually plateaus off at 96%. In Figure 6 (b), we show the time required to achieve a certain CA. From the overlap in network transmission and server processing, we observe that early termination can be achieved for a majority of query images. Figure 6 (b) can be interpreted as follows: e.g., for 60% of images, early termination can be achieved in approximately 2 seconds of system latency once the server completes its processing. Figure 7 illustrates the distribution of system latency when early termination is used. We observe that early termination can be achieved for most query images, and average latency can be reduced significantly.

Finally, in Figure 8, we study the average reduction in system latency with sequential transmission of feature descriptors at the highest CA point achieved (96%). The average bitrate and end-to-end latency can be computed as a weighted sum of different points on the curves in Figure 6. Compared to full data transmission, we achieve a 4× reduction in data, and a 2× reduction in system latency when feature data are incrementally transmitted. We discuss how we can further reduce system latency using spatial embedding.

4.2.2 *Base Layer/Enhancement Layer*

Progressive transmission with spatial embedding is illustrated in Figure 3. The base and enhancement layer data require different Inverted File Systems (IFS) on the server. Different Vocabulary Trees are trained for *Base Layer* embedded descriptors and full descriptors encompassing both *Base Layer* and *Enhancement Layer*. We store an IFS for *Base Layer* and *Base and Enhancement Layer* separately as illustrated in Figure 9.

We transmit the *Base Layer* data incrementally in blocks of 100 as discussed in Section 4.2.1. Once *Base Layer* data are received incrementally, the server can start the recognition process. The *Enhancement Layer* data are transmitted after all the *Base Layer* data are transmitted. If a match is found at any stage, the server early terminates and sends a response back to the client.

The *Base Layer* and *Enhancement Layer* data accuracy profiles are illustrated in Figure 10. As expected, we note that maximum *Base Layer* CA (92%) is lower than maximum *Enhancement Layer* CA (96%) as *Base Layer* data are less discriminative. At low rates, *Base Layer* data provides a better trade-off in bitrate and CA. By incrementally transmitting *Base Layer* data, followed by *Enhancement Layer* data, we obtain a further 10-20% decrease in average bitrate and system latency, as shown in Figure 8.

4.3 Comparisons to SIFT and JPEG

Figure 11 compares schemes based on CHoG, SIFT and JPEG. For the JPEG scheme, the bitrate is varied by changing the quality of compression. The compressed image is transmitted over the network, and all processing is done on the server. We observe that the performance of the JPEG scheme rapidly deteriorates at low bitrates. The performance suffers at low bitrates as the interest point detection fails due to JPEG compression artifacts.

For the SIFT scheme, each descriptor is transmitted uncompressed as 1024 bits (128 dimensions × 8 bits/dimension).

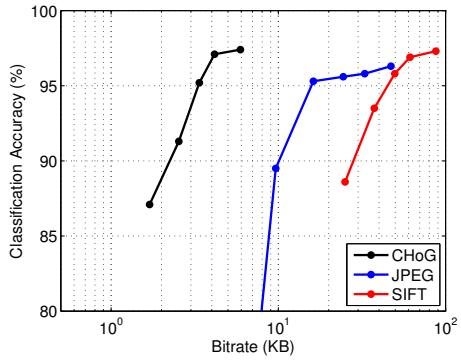


Figure 11: Bitrate comparisons of different schemes. CHoG descriptor data are an order of magnitude smaller compared to transmitting JPEG images or uncompressed SIFT descriptors.

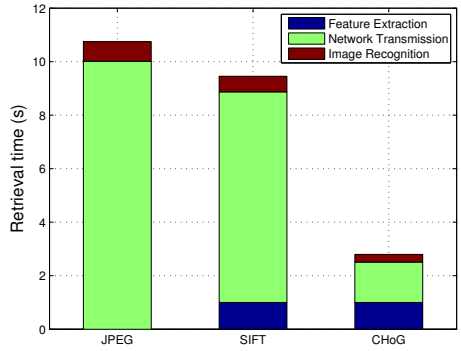


Figure 12: End-to-end latency for different schemes. Compared to SIFT and JPEG schemes, we achieve approximately 4× reduction in average system latency using progressive transmission of CHoG descriptors.

We sweep the CA-bitrate curve by varying the number of descriptors transmitted. We note that transmitting uncompressed SIFT data is almost always more expensive than transmitting JPEG compressed images. We observe that the amount of data for CHoG descriptors are an order of magnitude smaller than JPEG images or SIFT descriptors.

Finally, in Figure 12, we study the average end-to-end latency at the highest accuracy point for the different schemes. For the JPEG scheme, there is no processing on the client. For the SIFT and CHoG schemes, approximately 1 second is spent extracting features on the mobile client. For the SIFT scheme, we transmit descriptors incrementally in blocks of 100. For the CHoG scheme, we transmit *Base Layer* data incrementally in blocks of 100, followed by *Enhancement Layer* data. We achieve approximately 4× reduction in system latency with CHoG descriptors compared to JPEG images or uncompressed SIFT descriptors.

5. CONCLUSION

We demonstrate how network latency can be reduced significantly using rate-scalable CHoG descriptors. We propose two progressive transmission schemes, one based on incrementally sending descriptors, and another based on spatial embedding of descriptor data. Progressive transmission reduces the average amount of transmitted CHoG feature data by 4× and enables early termination on the server, thus reducing end-to-end system latency. With progressive transmission of CHoG, we are able to reduce network latency to ∼1 second in a 3G network. We report a 4× decrease in end-to-end system latency compared to transmitting uncompressed SIFT descriptors or JPEG images.

6. REFERENCES

[1] *Google Goggles*. `http://www.google.com/mobile/goggles/`.

[2] *Nokia Point and Find*. `http://www.pointandfind.nokia.com`.

[3] H. Bay, A. Ess, T. Tuytelaars, and L. V. Gool. Speeded-up robust feature. *Computer Vision and Image Understanding*, 110(3):346–359, 2008.

[4] V. Chandrasekhar, M. Makar, G. Takacs, D. Chen, S. S. Tsai, N. M. Cheung, R. Grzeszczuk, Y. Reznik, and B. Girod. Survey of SIFT Compression Schemes. In *Proc. of International Mobile Multimedia Workshop (IMMW), IEEE International Conference on Pattern Recognition (ICPR)*, Istanbul, Turkey, August 2010.

[5] V. Chandrasekhar, Y. Reznik, G. Takacs, D. M. Chen, S. S. Tsai, R. Grzeszczuk, and B. Girod. Study of Quantization Schemes for Low Bitrate CHoG descriptors. In *Proc. of IEEE International Workshop on Mobile Vision (IWMV)*, San Francisco, California, June 2010.

[6] V. Chandrasekhar, G. Takacs, D. M. Chen, S. S. Tsai, R. Grzeszczuk, and B. Girod. CHoG: Compressed Histogram of Gradients - A low bit rate feature descriptor. In *Proc. of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Miami, Florida, June 2009.

[7] D. M. Chen, S. S. Tsai, R. Vedantham, R. Grzeszczuk, and B. Girod. *CD Cover Database - Query Images*, April 2008. http://vcui2.nokiapaloalto.com/ dchen/cibr/testimages/.

[8] M. A. Fischler and R. C. Bolles. Random Sample Consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Communications of ACM*, 24(6):381–395, 1981.

[9] H. Jegou, M. Douze, and C. Schmid. Hamming embedding and weak geometric consistency for large scale image search. In *Proc. of European Conference on Computer Vision (ECCV)*, Berlin, Heidelberg, 2008.

[10] D. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[11] K. Mikolajczyk and C. Schmid. Performance evaluation of local descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(10):1615–1630, 2005.

[12] D. Nistér and H. Stewénius. Scalable recognition with a vocabulary tree. In *Proc. of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, New York, USA, June 2006.

[13] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman. Lost in quantization - improving particular object retrieval in large scale image databases. In *Proc. of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Anchorage, Alaska, June 2008.

[14] G. Takacs, V. Chandrasekhar, D. M. Chen, S. S. Tsai, R. Grzeszczuk, and B. Girod. Unified Real-time Tracking and Recognition with Rotation Invariant Fast Features. In *Accepted to IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, SFO, California, June 2010.

[15] S. S. Tsai, D. M. Chen, V. Chandrasekhar, G. Takacs, N.-M. Cheung, R. Vedantham, R. Grzeszczuk, and B. Girod. Mobile Product Recognition. In *Submitted to ACM Multimedia (ACM MM)*, Florence, Italy, October 2010.

[16] S. S. Tsai, D. M. Chen, J. Singh, and B. Girod. Rate-efficient, real-time CD cover recognition on a camera-phone. In *Proc. of ACM Multimedia (ACM MM)*, Vancouver, British Columbia, Canada, October 2008.

[17] S. S. Tsai, D. M. Chen, G. Takacs, V. Chandrasekhar, J. P. Singh, and B. Girod. Location coding for mobile image retreival systems. In *Proc. of International Mobile Multimedia Communications Conference (MobiMedia)*, London, UK, September 2009.

[18] S. Winder, G. Hua, and M. Brown. Picking the best daisy. In *Proc. of Computer Vision and Pattern Recognition (CVPR)*, Miami, Florida, June 2009.