A Study of Clustering Algorithms and Validity for Lossy Image Set Compression

Anthony Schmieder[1], Howard Cheng[1], and Xiaobo Li[2]
[1]Department of Mathematics and Computer Science, University of Lethbridge, Canada
[2]Department of Computing Science, University of Alberta, Canada

Abstract—*A hierarchical lossy image set compression algorithm (HMST$_a$) has recently been proposed for lossy compression of image sets. It was shown that this algorithm performs well when an image set contains well separated clusters of similar images. As a result, if one applies the HMST$_a$ algorithm after a clustering algorithm has been applied, the compression performance depends on the quality of the partition. In this paper, we examine a number of well-known hierarchical clustering methods and cluster validity measures, and their relationships to the compression performance of HMST$_a$. This relationship can be used as a component in a fully automated image set compression algorithm. We also briefly examine the merit of using different compression schemes depending on the compactness of the cluster in order to reduce computational complexity.*

Keywords: image set compression, clustering, cluster validity, minimum spanning tree.

1. Introduction

Traditional image compression algorithms for individual images, such as predictive coding and transform coding, have been shown to effectively reduce coding, inter-pixel, and psycho visual redundancy within an image [1]. Image sets, however, may contain inter-image redundancy, or "set redundancy" [2], which are not reduced by these algorithms. A number of algorithms have been proposed to address this issue [3], [4], [5], [2], [6], [7], [8], [9].

The MST$_a$ scheme [4], [5] computes a global average image of the entire image set, and forms a complete graph in which the vertices are the images (including the average) and the edges have weights being the cost of compressing the corresponding difference images. A minimum spanning tree (MST) is computed and the difference images associated to the edges in the MST are compressed. It was shown that this scheme works well for images that are very similar to each other. In cases where images in a set form well separated clusters of similar images, the global average may not be representative of any image in the set. The Hierarchical Minimum Spanning Tree (HMST$_a$) algorithm [9] partitions a set of images into clusters of similar images, and applies the MST$_a$ scheme on each of the clusters. The MST$_a$ scheme is then applied again to the set containing the average image of each cluster. The performance of the HMST$_a$ algorithm depends strongly on the quality of the partition of the images. Our ultimate goal in this research is to obtain an automatic lossy image set compression algorithm. The algorithm should automatically find an appropriate partition of the data and apply the HMST$_a$ algorithm.

In this paper, we examine a number of well-known hierarchical clustering algorithms and cluster validity measures, as well as their relationships to compression performance by HMST$_a$. We also briefly examine the merit of using different compression strategies depending on the compactness of the cluster. If the difference in compression performance between different schemes is small for compact clusters, one may choose a faster non-optimal algorithm without sacrificing much in compression performance. The results from this study are key components in an automatic image set compression algorithm.

2. Previous Works

Karadimitriou and Tyler proposed the Centroid "set mapping" scheme to reduce "set redundancy" for lossless compression [2], [6]. The Centroid scheme involves computing an average image for a set of similar images, calculating the difference between the average image and each image in the set, and coding the average image and the difference images. The corresponding lossy algorithm gave significant improvement in compression ratios compared to compressing individual images. However, the images in the set must be quite similar for the Centroid scheme to perform well, and image sets that contain dissimilar images were not considered.

The minimum spanning tree (MST) set mapping strategy, proposed by Nielsen and Li, is based on a graph data structure [7]. A complete graph is constructed, using images as the vertices and the root-mean-square error (RMSE) between adjacent images as the edge weights. An MST for the graph is calculated, and one image is chosen as the root. The root image and difference images represented by the edges with the lowest total cost are encoded using lossy JPEG2000 [10]. The results of these experiments showed a clear improvement in average distortion (RMSE) when using the MST scheme over the "traditional" scheme of compressing each image individually, especially at lower bitrates. These experiments focused on sets of similar images, and did not examine performance on sets containing dissimilar images. Other authors have applied a similar strategy to specific applications such as object movies [3], multiview video coding [11], multispectral images [12], and map images [13]. However, the image sets in these applications are known to be similar and the image sets are usually small.

Gergel *et al.* built upon this work with the MST$_a$ scheme [4], [5]. An MST is computed on a complete graph that includes a zero image and an average image, using RMSE as edge weight. The MST$_a$ scheme is a unified framework that adaptively chooses the best scheme among the traditional, Centroid, MST, and other schemes. The MST$_a$ scheme is shown to be highly effective, outperforming the other schemes in many cases. Schmieder *et al.* applied a clustering algorithm and applied the MST$_a$ algorithm first to each cluster and then to the cluster averages [9]. The HMST$_a$ algorithm was effective when the image set contains a number of clusters of very similar images, and the clustering algorithm accurately identifies the partition.

3. Approach

Our general approach is to apply a hierarchical clustering algorithm to produce a sequence of partitions and to select the best partition from this sequence. Of course, one can simply apply the HMST$_a$ algorithm to each partition, observe the result, and choose the partition that gave the best compression performance. However, the computational cost of the HMST$_a$ algorithm (including the compression of difference images) is high, and it is preferable to minimize the number of times it is invoked. Intuitively, the performance of the HMST$_a$ algorithm improves as the partition gets "better." There are a number of cluster validity measures that have been used to measure the quality of a partition. Therefore, we will examine the use of these validity measures to see how well they relate to the actual compression performance. Since these validity measures are much easier to compute, they can be used to choose the best partition provided that the validity measure has a strong correlation to the compression performance.

For images that are very similar, the Centroid scheme [2], [6] may compress an image set very well, with the additional advantage that it is significantly faster than the MST$_a$ algorithm. One may be interested in applying the Centroid scheme instead when an image cluster is "compact enough." We will also examine the effectiveness of a number of compression schemes for compact clusters in our experiments.

3.1 Cluster Analysis

For the purpose of cluster analysis, we represent each $M \times N$ image I_i in the image set S as an MN-dimensional vector whose components are the intensity values at each pixel. Each of these vectors is considered a pattern. Given two images I_i and I_j, we also define the root-mean-square error (RMSE) as

$$RMSE(I_i, I_j) = \sqrt{\frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} (I_i(x,y) - I_j(x,y))^2} \tag{1}$$

The RMSE is used as a dissimilarity measure [14] in the clustering algorithms. For the remainder of this paper, the notation $d(i,j) = RMSE(I_i, I_j)$ will denote the dissimilarity between I_i and I_j.

3.2 Cluster Validity Measures

Cluster validity measures can be used to evaluate how well a partition "fit" the given data. Since we are interested in an automatic image set compression algorithm, we only consider "internal" validity measures [14] which do not require a priori knowledge on the image set. In this study, we consider the Variance Ratio Criterion (VRC) [15], the C-index [16], and the Silhouette index [17]. The VRC and C-index were listed among the best-performing measures in a study of 30 internal validity measures as stopping rules of clustering algorithms by Milligan and Cooper [18].

We assume that $S = \{I_1, \ldots, I_n\}$ is an image set of n images, and it has been partitioned into a set of clusters $\mathcal{C} = \{C_1, \ldots, C_m\}$. It is assumed that the partition is not trivial; that is, $1 < m < n$.

3.2.1 Variance Ratio Criterion (VRC)

The Variance Ratio Criterion (VRC) is based on notions used in analysis of variance in statistics [15]. This measure is based on the idea that an ideal partition will have patterns in the same cluster being less dissimilar than patterns in different clusters.

First, the *within group sums of squares* (WGSS) is defined as

$$\text{WGSS} = \sum_{g=1}^{m} \left(\frac{1}{|C_g|} \sum_{I_i, I_j \in C_g, i<j} d(i,j)^2 \right), \tag{2}$$

and the *between group sum of squares* (BGSS) is defined as

$$\text{BGSS} = \frac{1}{n} \sum_{1 \le i < j \le n} d(i,j)^2 - \text{WGSS}. \tag{3}$$

Then, the VRC is defined as

$$\text{VRC} = \frac{\text{BGSS}}{m-1} / \frac{\text{WGSS}}{n-m}. \tag{4}$$

Intuitively, the WGSS measures the dissimilarity within clusters while the BGSS measures the dissimilarity between different clusters. Thus, a higher value of VRC indicates a better partition.

3.2.2 C-index

The C-index is defined as

$$C = \frac{d_w - \min(d_w)}{\max(d_w) - \min(d_w)}, \tag{5}$$

where

$$d_w = \sum_{g=1}^{m} \sum_{I_i, I_j \in C_g} d(i,j), \tag{6}$$

and $\min(d_w)$ ($\max(d_w)$) is the sum of the smallest (largest) k values of $d(i,j)$ ($1 \leq i < j \leq n$) with k being the number of terms in the sum (6) [16]. Intuitively, a better partition would minimize within cluster "scatter" and result in a smaller value for the C-index.

3.2.3 Silhouette Index

The Silhouette index was intended to be used as a graphical aid to cluster validation, but its value can also be used as a validity measure [17].

The silhouette index for a single pattern I_i in cluster C_k is defined as:

$$s(I_i) = \frac{b(I_i) - a(I_i)}{\max(a(I_i), b(I_i))}, \tag{7}$$

where $a(I_i)$ is the average dissimilarity of I_i in its cluster, and $b(I_i)$ is the minimum average distance from I_i to patterns in other clusters:

$$a(I_i) = \frac{1}{|C_k| - 1} \sum_{I_j \in C_k} d(i,j), \tag{8}$$

$$b(I_i) = \min_{1 \leq g \leq m, g \neq k} \left\{ \frac{1}{|C_g|} \sum_{I_j \in C_g} d(i,j) \right\}. \tag{9}$$

If I_i is in a singleton cluster then $s(I_i) = 0$. The global silhouette index is simply the average of the silhouette index for the entire pattern set. Intuitively, the silhouette index measures how well a pattern "fits" in its assigned cluster by comparing its dissimilarity to patterns within and outside its own cluster. A higher value of global silhouette index indicates a better partition.

3.3 Hierarchical Clustering Algorithms

A *hierarchical clustering* is a sequence of partitions of the image set into clusters, such that the first partition of the sequence contains only singleton clusters (*disjoint partition*), and the last partition contains a single cluster of all images (*cojoint partition*) [14]. Each partition in the sequence is *nested into* the next partition. That is, each cluster in the partition is a subset of some cluster in the next partition. For this study, we examine *agglomerative* hierarchical clustering algorithms which begins with the disjoint partition and ends with the cojoint partition. We examine the same four clustering algorithms in Milligan's study of cluster validity measures [19].

3.3.1 Single Link Method

At each step of the single link method [14], two images I_i and I_j from different clusters are chosen so that $d(i,j)$ is minimum among all such image pairs. The two clusters are then merged into one cluster. Note that this is equivalent to Kruskal's algorithm for constructing a minimum spanning tree for a graph.

3.3.2 Complete Link Method

In the complete link method [14], a graph is initially formed so that the images are the vertices and there are no edges between any vertices. The dissimilarities $d(i,j)$ are examined in increasing order, and the corresponding edges (I_i, I_j) are added to the graph incrementally. Whenever a new clique in the graph is formed from the vertices in two existing clusters, the clusters are merged into one.

3.3.3 UPGMA

In the Pair Group Method using Unweighted Averages (UPGMA), the dissimilarity between two clusters is the average dissimilarity between pairs of images with one image in each cluster. At each step, the two remaining clusters with the smallest dissimilarity are merged into one.

3.3.4 Ward's Method

At each step of Ward's method, two clusters are chosen so that if the clusters were merged, the increase in error sum of squares (sum of squares of dissimilarity between each image and its cluster average) is minimum.

3.4 Cluster Compactness

In this study, we consider the diameter and radius of a cluster as measures of its compactness. Let $I_a = I_{n+1}$ be the average image of the set $S = \{I_1, \ldots, I_n\}$. We define the diameter and radius as:

$$D(S) = \max_{1 \leq i < j \leq n} d(i,j) \tag{10}$$

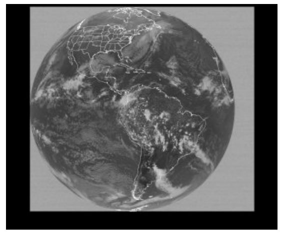
$$R(S) = \max_{1 \leq i \leq n} d(i, n+1). \tag{11}$$

A smaller value for diameter or radius indicates that the cluster is more compact.

4. Experiments

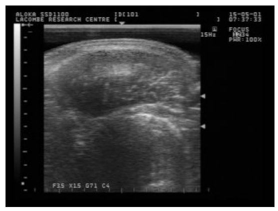
We use the same five test image sets used in previous image set compression studies [4], [5]: Galway [20], GOES [21], Joe [22], Pig, and Combination. Sample images appear in Figure 4. The Galway set contains 28 webcam images of a street in Galway City, Ireland that were taken within a few hours under similar weather and lighting conditions. The GOES set contains 128 satellite images of the Earth. The Joe set contains 162 webcam images of an outdoor scene that were taken under varying weather and lighting conditions. The Pig set contains 304 ultrasound images of pig rib cages.

(a) Galway



(b) GOES

(c) Joe



(d) Pig

Fig. 1: Sample images from each set.

Finally, the Combination set contains 57 images: 28 from the Galway set, and 29 from the Pig set.

4.1 Cluster Validity Measures

We are interested in evaluating how well various cluster validity measures predict the compression performance of a particular partition. Different hierarchical clustering algorithms are applied to each image set. The HMST$_a$ algorithm and various validity measures are applied to the members of the sequence of partitions produced by the clustering algorithms. To measure compression performance for the lossy HMST$_a$ algorithm, we perform compression at a fixed compression ratio (1:10, 1:30, 1:50, 1:70, and 1:90) and measure the average distortion (RMSE) between the original images and the decompressed images. A lower average distortion indicates better compression performance. The Pearson correlation coefficient (r) is computed to determine how well each validity measure can predict the compression performance. For the VRC and Silhouette measures (larger value indicates better partition), an r value close to -1 indicates the measure correlates well with the compression performance. For the C-index measure, an r value close to 1 indicates strong correlation.

Tables 1 through 5 show the correlation coefficients in our experiments. In all of our experiments, the UPGMA and Ward's methods for clustering produce the same partitions. These two methods generally produce poor partitions on the image sets—each partition consists only of one large cluster and the remaining images in singleton clusters. The validity measures are generally not well-suited for partitions with many singleton clusters, so none of the validity measures work well in these cases.

We see that the Silhouette index generally has a higher correlation with the compression performance of HMST$_a$ than VRC or C-index, especially at higher compression ratios. In many cases, the correlation of C-index (and sometimes VRC) to the compression performance has the wrong sign. This happens mainly because the difference in the compression performance among different partitions is relatively small, and a few outliers may affect the correlation significantly. Sometimes the correlation has the wrong sign even if the most "valid" partition indeeds yield the best compression performance. Nevertheless, there are a few instances where using these measures to predict compression performance may give the opposite effect. The correlation of the Silhouette measure with compression performance has the correct sign and is fairly close to -1 in most cases, except for the Galway image set. For this image set, the correlation is still reasonable when the complete link clustering method is used. For the Galway image set, the bad correlation values are generally caused by a few outliers. Choosing a partition based on the Silhouette measure will generally give one of the best partitions even in the single link case.

From the experimental results, we may conclude that the Silhouette index is the best among the three validity measures studied in its ability to predict the compression performance of the HMST$_a$ algorithm. In addition, the Silhouette index performs better in conjunction with the complete link method than other clustering methods.

4.2 Performance on Compact Clusters

We now evaluate the effectiveness of various set image compression schemes on compact clusters. We examine the traditional scheme (each image is compressed separately), the Centroid scheme [2], [6], the MST scheme [7], and the MST$_a$ scheme [4], [5]. We examine the average distortion between the original and decompressed images at various compression ratio for each scheme. Although the Centroid scheme is computationally simpler, our experiments show that the MST$_a$ scheme still provides a significant improvement to the compression performance even for relatively compact clusters. A sample plot of the experimental results is shown in Figure 2. Even for a very compact image set, the Centroid scheme can be far from optimal. For example, Figure 3 shows a set with images I_i and I_j (other images omitted) and centroid $I_{n+1} = I_a$ where $d(i,j) < d(n+1,j)$. The Centroid scheme would select the edge (I_a, I_j) instead of (I_i, I_j). Of course, if $R(S)$ is very small, then $d(I_{n+1}, I_j) < R(S)$ implies that the penalty in compression performance may be small while the computational complexity for the Centroid scheme is much lower. However, we have not encountered any non-artificial image set in which the MST$_a$ algorithm does not outperform the Centroid scheme significantly even for compact clusters. Thus, we can conclude from our experimental results that the MST$_a$ algorithm should be used regardless of the compactness of clusters unless computational complexity is a significant concern.

Table 1: Galway image set.

	VRC			C-index			Silhouette		
Ratio	C. Link	S. Link	UPGMA/ Wards	C. Link	S. Link	UPGMA/ Wards	C. Link	S. Link	UPGMA/ Wards
1:10	-0.6536	-0.3810	0.0752	-0.8427	-0.3534	-0.6899	-0.7917	-0.6567	-0.9325
1:30	0.0926	0.3102	0.0825	0.1346	0.1310	0.4895	-0.1561	0.4958	0.9100
1:50	0.2112	0.4236	0.0944	0.2475	0.1072	0.5061	-0.1748	0.5397	0.9124
1:70	0.0137	0.4109	0.5643	-0.0103	-0.0719	0.0763	-0.4350	0.3130	0.7911
1:90	-0.2864	0.0480	0.6444	-0.4122	-0.3477	-0.7992	-0.7701	-0.3596	-0.4773

Table 2: GOES image set.

	VRC			C-index			Silhouette		
Ratio	C. Link	S. Link	UPGMA/ Wards	C. Link	S. Link	UPGMA/ Wards	C. Link	S. Link	UPGMA/ Wards
1:10	-0.0675	-0.0444	-0.1911	-0.1851	0.6444	0.2949	-0.2052	0.3922	0.1977
1:30	0.0882	0.1194	0.7085	-0.9077	-0.7868	-0.8146	-0.9329	-0.8695	-0.6762
1:50	-0.0202	-0.0284	0.6802	-0.9081	-0.7277	-0.7796	-0.9310	-0.9516	-0.6142
1:70	-0.0921	-0.1244	0.6898	-0.8698	-0.6653	-0.7781	-0.8940	-0.9167	-0.6173
1:90	-0.1032	-0.1408	0.6807	-0.8324	-0.6199	-0.7656	-0.8495	-0.8757	-0.6007

Table 3: Joe image set.

	VRC			C-index			Silhouette		
Ratio	C. Link	S. Link	UPGMA/ Wards	C. Link	S. Link	UPGMA/ Wards	C. Link	S. Link	UPGMA/ Wards
1:10	-0.5062	-0.1116	0.0082	-0.5766	-0.3437	-0.4125	-0.5035	-0.0951	-0.4233
1:30	-0.0134	0.8011	0.4579	-0.5312	-0.2539	-0.8102	-0.9352	-0.4948	-0.7743
1:50	-0.2245	0.8221	0.2778	-0.6541	-0.3958	-0.7366	-0.9413	-0.6739	-0.6405
1:70	-0.2284	0.8743	0.4072	-0.6610	-0.4151	-0.7189	-0.9307	-0.6754	-0.7473
1:90	-0.1766	0.8869	0.4284	-0.6222	-0.4100	-0.6983	-0.9048	-0.6779	-0.7700

Table 4: Pig image set.

	VRC			C-index			Silhouette		
Ratio	C. Link	S. Link	UPGMA/ Wards	C. Link	S. Link	UPGMA/ Wards	C. Link	S. Link	UPGMA/ Wards
1:10	0.5320	0.4608	0.4742	0.0378	-0.2395	0.36567	0.4167	-0.2623	0.4030
1:30	-0.7571	-0.4566	-0.7811	-0.5643	-0.5316	-0.3666	-0.7407	-0.8732	-0.5023
1:50	-0.7196	-0.3484	-0.8187	-0.5752	-0.6594	-0.3153	-0.7823	-0.8917	-0.5249
1:70	-0.6178	-0.3121	-0.8066	-0.6769	-0.6935	-0.2858	-0.8009	-0.8612	-0.4930
1:90	-0.5579	-0.2716	-0.8097	-0.7020	-0.7168	-0.2887	-0.8338	-0.8172	-0.4956

5. Conclusion

In this paper, we showed that the Silhouette index when combined with hierarchical clustering methods perform quite well in predicting the compression performance of the HMST$_a$ algorithm. We plan to incorporate this measure into an image set compression algorithm to automatically choose the best partition of the image set. We also showed that even for compact clusters, the MST$_a$ algorithm may still perform significantly better than the computationally simpler Centroid scheme. Thus, it is not advisable for an automatic image set compression algorithm to incorporate the Centroid scheme unless computational complexity is a significant concern.

Acknowledgements

We would like to gratefully acknowledge Dr. Alan Tong of Lacombe Research Centre, Agriculture and Agri-Food Canada for making the Ultrasound images and data available to this study.

References

[1] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 3rd ed. Prentice Hall, 2007.

[2] K. Karadimitriou, “Set redundancy, the enhanced compression model, and methods for compressing sets of similar images,” Ph.D. dissertation, Louisiana State University, 1996.

[3] C.-P. Chen, C.-S. Chen, K.-L. Chung, H.-I. Lu, and G. Tang, “Image set compression through minimal-cost prediction structures,” in *IEEE International Conference on Image Processing*, 2004, pp. 1289–1292.

[4] B. Gergel, “Automatic compression for image sets using a graph theoretical framework,” Master’s thesis, University of Lethbridge, 2007.

[5] B. Gergel, H. Cheng, C. Nielsen, and X. Li, “A unified framework for image set compression,” in *Proceedings of the 2006 International Conference on Image Processing, Computer Vision, & Pattern Recognition*, 2006, pp. 417–423.

[6] K. Karadimitriou and J. M. Tyler, “The centroid method for compressing sets of similar images,” *Pattern Recognition Letters*, vol. 19, no. 7, pp. 585–593, 1998.

[7] C. Nielsen and X. Li, “MST for lossy compression on image sets,” in *Proceedings of the Data Compression Conference*, 2006, p. 463.

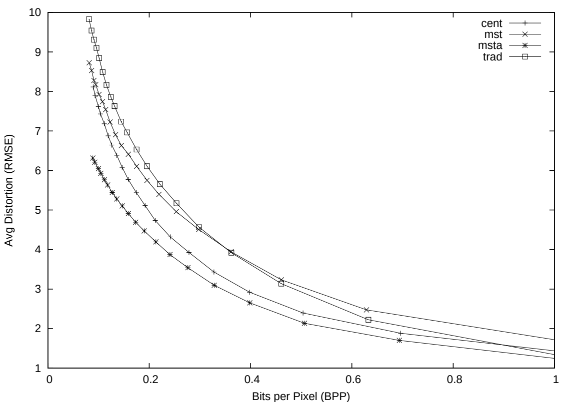
[8] C. Nielsen, X. Li, and K. Abma, “Methods of grouping similar images for compression coding.” in *Proceedings of The 2005 International Conference on Computer Vision*. CSREA Press, 2005, pp. 93–99.

[9] A. Schmieder, H. Cheng, B. Gergel, and X. Li, “Hierarchical Minimum Spanning Trees for Lossy Image Set Compression,” in *Proceedings of*

Table 5: Combination image set.

	VRC			C-index			Silhouette		
Ratio	C. Link	S. Link	UPGMA/ Wards	C. Link	S. Link	UPGMA/ Wards	C. Link	S. Link	UPGMA/ Wards
1:10	0.2286	0.0993	-0.9058	-0.7093	-0.2185	0.8947	-0.5246	-0.7962	0.7390
1:30	-0.5061	-0.3884	-0.7359	-0.2008	0.0308	0.8275	-0.9508	-0.9655	0.6263
1:50	-0.4685	-0.3659	-0.6662	-0.2031	0.0534	0.7954	-0.9628	-0.9586	0.6130
1:70	-0.4123	-0.3478	-0.5842	-0.2361	0.0668	0.7302	-0.9656	-0.9520	0.5577
1:90	-0.3910	-0.3503	-0.5514	-0.2337	0.0838	0.7110	-0.9667	-0.9491	0.5492

Fig. 2: Compression performance of various set compression schemes for a cluster in the Combination image set ($R(S) = 14.3$, $D(S) = 21.4$).



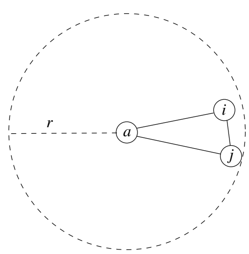


Fig. 3: Image set where Centroid scheme is suboptimal.

the 2008 International Conference on Image Processing, Computer Vision, & Pattern Recognition (IPCV'08), 2008, pp. 57–63.

[10] T. J. Committee, "JPEG 2000," http://www.jpeg.org/jpeg2000/index.html.

[11] D.-X. Li, W. Zheng, X.-H. Xie, and M. Zhang, "Optimising interview prediction structure for multiview video coding with minimum spanning tree," *Electronics Letters*, vol. 43, no. 23, pp. 1269–1271, November 2007.

[12] S. R. Tate, "Band ordering in lossless compression of multispectral images," *IEEE Transactions on Computers*, vol. 46, no. 4, pp. 477–483, April 1997.

[13] P. Kopylov and P. Franti, "Compression of map images by multilayer context tree modeling," *IEEE Transactions on Image Processing*, vol. 14, no. 1, pp. 1–11, January 2005.

[14] A. K. Jain and R. C. Dubes, *Algorithms for Clustering Data*. Prentice Hall, 1988.

[15] T. Caliński and J. Harabasz, "A Dendrite Method for Cluster Analysis," *Communications in Statistics*, vol. 3, pp. 1–27, 1974.

[16] L. J. Hubert and J. R. Levin, "A General Statistical Framework for Assessing Categorical Clustering in Free Recall," *Psychological Bulletin*, vol. 83, pp. 1072–1080, 1976.

[17] P. J. Rousseeuw, "Silhouettes: A Graphical Aid to the Interpretation and Validation of Cluster Analysis," *Journal of Computational and Applied Mathematics*, vol. 20, pp. 53–65, 1987.

[18] G. W. Milligan and M. C. Cooper, "An Examination of Procedures for Determining the Number of Clusters in a Data Set," *Psychometrika*, vol. 50, pp. 159–179, 1985.

[19] G. W. Milligan, "A Monte Carlo Study of Thirty Internal Criterion Measures for Cluster Analysis," *Psychometrika*, vol. 46, pp. 187–199, 1981.

[20] "Galway.net webcams," http://www.galway.net/about/webcam/.

[21] "GOES project science," http://rsd.gsfc.nasa.gov/goes.

[22] "JoeTourist Weather Station," http://wx.joetourist.net/webcam/Default.htm.