Codebook Design for Vector Quantisation Using Multiobjective Genetic Algorithms

Rajeev Kumar

Department of Computer Science & Information Systems
Birla Institute of Technology & Science, Pilani – 333 031, India
rajeevk@bits-pilani.ac.in

Abstract. In this paper, we present a novel approach for codebook design using multiobjective genetic optimization. We partition the pattern space into perceptive classes, which are then clustered into hyperspheres based on multiobjective criteria. Multiobjective genetic algorithms perform optimisation on a vector space of objectives and are able to explore the search space for a set of equally viable and equivalent partitions of the pattern space. One of the objectives searches for compact solutions within a class, which in turn yields the least MSE distortion along with better perceptive quality.

1 Introduction

Vector Quantisation (VQ) reduces the bit rate required to represent signals by utilising a codebook, which stores a fixed set of vectors, and then coding the decomposed signal by using the index/address of the codebook. For image compression, the vector corresponds to a small subimage or block, and VQ treats the entire subimage as a single entity and quantises it by reducing the total number of bits required to represent the subimage.

Among the various algorithms proposed for construction of VQ, most of them are *essentially* the clustering algorithms [1]. Most traditional clustering algorithms rely on some *similarity* measure (which is usually ill defined), and the resulting clusters depend directly on the judgment of what are and what are not nominally identical patterns – they may also fail to converge to a local minimum. There is one open issue regarding the convergence properties among the various algorithms proposed for VQ [2]. With improper initial selection of codevectors, the codebook design algorithm may converge to a local minimum. On the other hand, construction of codebook demands that the clusters/partitions should be generated on the basis of fitness for purpose rather than emerging as some *implicit* property of the clustering algorithm. The partitions should be *explicitly* optimized for the desired objective(s) with the additional constraints, if any. In this work, we attempt to adopt this line of research.

2 Multiobjective Genetic Optimisation for Codebook

The design of a VQ may be stated as finding an optimal set of codevectors to represent the images generated by a particular source. The VQ may also be generated using the known distribution of the source/training images. However, in the current work, we generate VQ from the training image(s). We generate the codebook in two steps: Classification followed by a Partitioning using multiobjective genetic optimization for each class. In our approach, the pattern space, *i.e.*, the codevectors extracted from the training image(s) are first classified into distinct classes using Classified VQ [1] and then clustered to a set of hyperspheres using multiobjective genetic algorithm for a set of objectives [3]. The design of a codebook using training images may be thought as partitioning of a *k*-dimensional pattern space into hyperspheres such that the centroids of the blobs in each of the hyperspheres form the codevector.

We have identified the following set of objectives for partitioning with a *sole* aim of minimising the MSE in order to get an *optimal* codebook.

- Minimise MSE of included patterns aims at minimising the average MSE of the vectors in that cluster to the centroid of the cluster.
- Minimise the surface area attempts to produce compact solutions.
- Maximise the included patterns aims to include within all the partitions as many training vectors as possible. Hopefully outliers within the pattern space can be excluded.
- Maximise the included patterns aims to include within all the partitions as many training vectors as possible.
- Minimise the maximum MSE across partitions is in accordance with the Partial Distortion Theorem.
- Minimise the number of hyperspheres aims at reducing the bit-rate. Alternatively, this objective can be withdrawn, and the number of clusters can be specified in advance based on some prior knowledge of the source.

All the objectives ensure a fair distribution of potential solutions. We do not scalarise the fitness of the solutions into a weighted sum of objective-values but compare on the scale of pareto-ranking/non-dominance. At convergence, any given one of these objectives cannot be improved without degrading at least one of the others leading to a *set* of equivalent solutions.

Exploring an NP-complete search space for a set of equally viable partitions of the pattern space is a complex optimisation. So we imposed additional constraints on decision variables, and added a few heuristics. We used the Pareto Converging Genetic Algorithm (PCGA) [4], which naturally performs good sampling of the solution space and ensures population advancement towards the pareto-front. The added advantages of PCGA strategy are that the algorithm does not need *a priori* knowledge of the objective space and ensures the population advancement towards convergence. From the obtained set of solutions, a small subsets based on sub-ranking of objectives from each class were picked for the final codebook.

3 Results and Discussion

Experiments on VQ codebook design have been carried out to investigate the proposed algorithm by comparing it with the well-established VQ algorithms. We have used Lena image and the other images of human faces for experiments. For quantifying the performance, we have used Mean Square Error (MSE) as the distortion measure and the Peak Signal-to-Noise Ratio (PSNR) for the signal.

Initially, we classified the codevectors taken from Lena image and did the clustering using GAs for each class separately. The computed MSE and PSNR were comparable. We repeated the experiments for other images and found consistently good results. The one major issue that concerns the codebook generation here is the proper selection of subcodebook sizes for each class. In this work, we empirically approximated the subcodebook sizes, nonetheless a comprehensive genetic search with proper set of objectives could look for such decisions. We also did the experiments for generating a global codebook for the images taken from similar source.

We are unable to include the detailed results due to limitation on number of pages; the results shall be presented during the workshop.

4 Conclusions

In this paper, we have presented a novel approach to codebook construction for vector quantisation using multiobjective genetic algorithms. The preliminary results have shown that the performance is comparable to other well-established approaches and we have got consistently good codebooks. The performance may be further improved by tuning the parameters needed for genetic optimization.

This approach is a *generic* framework where the clusters are *explicitly* optimized for a set of user-defined objectives. In the present work, the objectives were tuned to minimising the average MSE. Nonetheless, any other desired set of objectives could be employed to construct a codebook.

References

1. Gersho, A., Gray, R.M.: Vector Quantisation and Signal Compression. Kluwer, Boston MA (1992)
2. Kosmatopoulos, E.B., Christodoulou, M.A.: Convergence Properties of a Class of Learning Vector Quantisation Algorithms. IEEE Trans. Image Processing 5 (1996) 361-368
3. Kumar, R., Rockett, P.I.: Multiobjective Genetic Algorithm Partitioning for Hierarchical Learning of High Dimensional Pattern Spaces: A Learning Follows Decomposition Strategy. IEEE Trans. Neural Networks 9 (1998) 822-830
4. Kumar, R., Rockett, P.I.: Improved Sampling of the Pareto front in Multiobjective Genetic Optimisations by Neo-Stationary Evolution: a Pareto Converging Genetic Algorithm. Evolutionary Computation (in press)