

Chapter 12

Fast Vector Quantization Encoding Algorithms for Image Compression

Ahmed Swilem
Minia University, Egypt

ABSTRACT

Vector quantization (VQ) is a well-known compression method. In the encoding phase, given a block represented as a vector, searching the closest codeword in the codebook is a time-consuming task. In this paper, two fast encoding algorithms for VQ are proposed. To reduce the search area and accelerate the search process, the first algorithm utilizes three significant features of a vector that are, the norm, and two projection angles to two projection axes. The second algorithm uses the first two features as in the first algorithm with the projection value of the vector to the second projection axis. The algorithms allow significant acceleration in the encoding process. Experimental results are presented on image block data. These results confirm the effectiveness of the proposed algorithms.

1. INTRODUCTION

Thanks to the incredibly rapid advancement of the Internet, online information communication and human connection, where the delivery and storage of digital images of all kinds have become the main stream, seem to have been planning a more and more important part of our life. However, for massive quantities of multimedia data to get traveled online from place to place at least acceptable speed or to be stored without taking

up too much of the memory space, especially when the network bandwidth is limited with small hardware storage, there is nothing more important than a good digital image compression scheme. Image compression methods can be categorized as two types: lossless compression and lossy compression. In lossless compression scheme, the original image can be recreated exactly from the compressed data but lossy compression scheme must exist some distortion between the original image and the reconstructed image. So far, vector quantization (VQ) (Gray, 1984; Linde, Buzo, & Gray, 1980) has long been a well-celebrated

DOI: 10.4018/978-1-60960-563-6.ch012

lossy compression technique that guarantees the achievement of a satisfactory balance between image quality and compression ratio (Chen, Chang, & Hwang, 1998). At the same time, because of its simple and easy implementation, VQ has been very popular in a variety of research fields such as speech recognition and face detection (Garcia, & Tziritas, 1999). Even in real-time video-based events detection (Liao, Chen, Su, & Tyan, 2006) and the anomaly intrusion detection systems (Zheng, & Hu, 2006), VQ has been exploited recently to learn and collect some representative patterns and then to identify similar feature vectors or detect unusual activities. Next, the procedure of vector quantization will be interpreted from the aspect of image compression.

VQ can be defined as a mapping Q from a k -dimensional Euclidean space R^k to a finite set $C = \{c_1, \dots, c_N\}$ of vectors in R^k called the codebook. Each representative vector c_i in the codebook is called a codeword. Generally, VQ can be divided into three procedures: codebook design, encoding and decoding. The codebook design procedure is executed before the other two procedures for VQ. The goal of the codebook design is to construct a codebook C from a set of training vectors using clustering algorithms like the generalized Lloyd algorithm (GLA) (Linde, Buzo, & Gray, 1980). This codebook is used in both the image encoding/decoding procedures. In the encoding procedure, for each training vector q , we find the index i of the codeword c_i is the closest codeword to the vector q which gives minimum distortion and satisfy $d^2(q, c_i) < d^2(q, c_j), \{j = 1, 2, \dots, N; i \neq j\}$, where

$$d^2(q, c_i) = \sum_{j=1}^k (q_j - c_{ij})^2, \quad (1)$$

so the codeword c_i now represents the vector q . The decoding procedure is simply a table look-up

procedure that uses the received index i to deduce the reproduction codeword c_i , and then uses c_i to represent the input vector q .

Using the squared Euclidean distance criteria in (1), each distortion computation requires k multiplications and $2k - 1$ additions. For an exhaustive full search (FS) algorithm, encoding each input vector requires N distortion calculations and $N - 1$ comparisons. Therefore, it is necessary to perform kN multiplications and $(2k - 1)N$ additions and $N - 1$ comparisons to encode each input vector. When and/or k become larger, the computation complexity problem occurs for full codebook search. Many algorithms were proposed later in this field to reduce the computational calculation needed to accomplish the VQ technique. Some of these algorithms use one or more constraint inequality in the spatial domain to reduce the computational calculation needed to accomplish the VQ technique (Bake, Bae, & Sung, 1999; Cao, & Li 2000; Cardinal, 1999; Guan, & Kamel, 1992; Huang, Bi, Stiles, & Harris, 1992; Imamura, Swilem, & Hashimoto, 2003; Imamura, Swilem, & Hashimoto, 2004; Johnson, Ladner, & Riskin, 1996; Johnson, Ladner, & Riskin, 2000; Lai, & Lue, 1996; Lee, & Chen, 1994; Orchard, 1991; Swilem, Imamura, & Hashimoto, 2002; Swilem, Imamura, & Hashimoto, 2004; Wu, & Lin, 2000). Other algorithms exploit the topological structure of vectors to accomplish the same target (Lee, & Chen, 1995; Pan, Lu, & Sun, 2000; Song, & Ra, 2002; Swilem, Imamura, & Hashimoto, 2005).

An algorithm for fast nearest neighbor search presented by Orchard (Orchard, 1991) precomputes and stores the distance between each pair of codewords. Given an input vector q , the current best codeword c_b , and a candidate codeword c_j , if $d(q, c_j) \leq d(q, c_b)$, then $d(c_j, c_b) \leq 2d(q, c_b)$. Graphically, this constrains the search area within a sphere centered on the current best codeword, with a radius of twice the smallest distortion calculated so far. In (Huang, Bi, Stiles, & Harris, 1992), an additional constraint on codewords by

11 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage:

www.igi-global.com/chapter/fast-vector-quantization-encoding-algorithms/53177

Related Content

Mobile Computing: Technology Challenges, Constraints, and Standards

Anastasis A. Sofokleous, Marios C. Angelides and Christos N. Schizas (2009). *Handbook of Research on Mobile Multimedia, Second Edition* (pp. 820-828).

www.irma-international.org/chapter/mobile-computing-technology-challenges-constraints/21047

A Survey of Visual Traffic Surveillance Using Spatio-Temporal Analysis and Mining

Chengcui Zhang (2013). *International Journal of Multimedia Data Engineering and Management* (pp. 42-60).

www.irma-international.org/article/a-survey-of-visual-traffic-surveillance-using-spatio-temporal-analysis-and-mining/95207

Default Reasoning for Forensic Visual Surveillance based on Subjective Logic and Its Comparison with L-Fuzzy Set Based Approaches

Seunghan Han and Walter Stechele (2011). *International Journal of Multimedia Data Engineering and Management* (pp. 38-86).

www.irma-international.org/article/default-reasoning-forensic-visual-surveillance/52774

Transcoding vs. Scalability in Video Streaming for Heterogeneous Networks/Clients

Nicola Conci and Francesco G.B. De Natale (2009). *Multimedia Transcoding in Mobile and Wireless Networks* (pp. 92-124).

www.irma-international.org/chapter/transcoding-scalability-video-streaming-heterogeneous/27197

Multi-Sensor Motion Fusion Using Deep Neural Network Learning

Xinyao Sun, Anup Basu and Irene Cheng (2017). *International Journal of Multimedia Data Engineering and Management* (pp. 1-18).

www.irma-international.org/article/multi-sensor-motion-fusion-using-deep-neural-network-learning/187137