Efficient Chain-Code Encoding for Segmentation-Based Image Compression

Dan E. Tamir, Kim Phillips, and Abdul-razzak Abdul-karim

ABSTRACT

This paper presents a new and efficient method of encoding uniform image regions and lines. Regions and lines are obtained as the result of image segmentation, split and merge image compression, or as the output of line and polygon drawing algorithms. Lines and contours of uniform regions are encoded using chain-code. The chain-code is obtained in a way that is efficient with respect to bit-rate and produces lossless contour and line encoding. A lossy method for contour encoding is also presented.

A set of experiments to compare the performance of traditional chain-code contour encoding with the improved contour encoding is presented. The results show a reduction of about 50% in the bit-rate with no reconstruction error.

Efficient lossless chain-encoding of region boundaries

Simple modifications to the encoding can lead to a significant saving in bit rate. Pavlidis' proposes a relative chain-code with variable code per direction. The code used in this paper is a fixed length relative code with priority where a lower number denotes higher priority. This coding method is designed to increase code redundancy.

A cut of almost 50% in the chain-code can be achieved by modifying the traditional approach of contour tracing into a region boundary tracing method. The problem with contour following is that pixels on region contours are also on the boundaries between regions. Hence, the contour following algorithm encodes each boundary twice. The change in approach entails that chains do not describe closed objects. Therefore, a few adjustment to the encoding scheme are required. First, one can not use the condition that the coordinates of the initial pixel and the coordinates of the terminating pixel of a chain are the same, as the means to identify the termination of a chain. Rather, a chain is ended with a relative prioritized code denoting going back to the previous chain element. Second, this approach has an additional overhead due to the requirement to include the coordinates of the initial pixel. Now, the same region may be represented by more than one chain and hence more than one initial pixel. Another modification is that instead of attaching the information about a region to its chain-code; the information about all the regions in the image proceeds or succeeds the chain-codes of the regions.

The coding scheme has a built-in LZW compression. In a fork point, this algorithm checks the compression table. If possible it chooses a route that is an entry in the table in favor of a route that requires updating the table. A lossy option that produce smooth edges is also available.

Experiments and results

The different improvements described in this paper has been applied to four images. One of these images has been obtained from line and polygon drawing. Other images are "natural" images that have been passed through segmentation. Our findings are: 1) Unix compression of the regular chain-code achieves between 1.5 times compression to 10 times compression. 2) Boundary Tracing with lossless compression is better than contour tracing with lossless compression. It's effect is two times better than Unix compression. 3) Implementing a built-in LZW has a slight improvement in the compression. 4) The lossy compression increases the bit-rate without additional distortion.

Conclusion

The results of these experiments show an improvement of about 48% (average over the four images) in the size of the code with no additional distortion. Compression with distortion can only increase the compression while adding objective and subjective distortion. We have found that lossy compression of the code at 46% (average over the four images) compression has negligible negative effect on the subjective image quality.

1. Computer Science Program, Florida Institute of Technology, Melbourne, FL 32901; tamir@cs.fit.edu