Image comparison via edge maps using Normalized Compression Distance

By:
Dudi Cohen
Image comparison

- Image comparison is a method of computing how different or similar two or more images are from each other.

- There are multiple methods of image comparison, each method varies from the other mainly by two elements:
  - Image features being compared.
  - Feature comparison method.
Why use edge maps?

• In my project I suggest an image comparison method in which the compared feature is the image’s edge map.

• We would like to use this sort of method when:
  – Colors aren’t needed to be accounted for defining the similarity.
  – Edges and contours are a big factor in the images.
How to compare edge maps?

• The main problem with comparing edge maps is choosing the comparison method:
  – Edges can appear in different coordinates in the image.
  – Edges can appear in different orientations in the image.

• Methods such as least square or hamming distance will not give us the proper results.

• For the comparison method I’ve used Normalized Compression Distance.
Kolmogorov complexity

• Kolmogorov complexity is the length of the shortest binary program to create an object.

• For example to create a black grayscale image, the C program is:
  – For (int i=0 ;i<width;i++)
    for (int j=0;j<height;j++)
      I[i][j]=0;

• K(x|y): length of the shortest program p that creates x from an input y

• K(x): length of the shortest program p that creates x from an input y=the empty word.
Kolmogorov complexity (cont.)

• It is easy to see that we can use $K(x|y)$ as a method to compare the similarity of 2 objects.

• $K(x)$ is actually the ultimate compressed version of $x$: takes advantage of every redundancy in $x$.

• All the information in $x$ is concentrated in it’s shortest program.

• Unfortunately $K(x)$ isn’t computable, so we need to find a method to give us a good approximation.

• $K(x)$ uses redundancy, so why not use a compressor?
Normalized Compression Distance (NCD)

- Compressors use redundancy in an object to encode it in a shorter way. Therefore it seems natural to approximate \( K(x) \) by \( C(x) \)- the compressed version of \( x \):

\[
NCD(x, y) = \frac{C(xy) - \min\{C(x), C(y)\}}{\max\{C(x), C(y)\}}
\]

- If two objects are similar then the length of the compressed version of their union should be close to the length of their compressed version

- \( \text{NCD}(x,y) = 0 \); identity
- \( \text{NCD}(x,y) = 1 \); most different
“Support Tools”

• I needed to select an edge detector and an effective compressor:
  – Edge detection was done using the Canny edge detector.
  – Compression was accomplished by implementing a Universal Lossless Data Compression of ROBDD (Reduced Ordered Binary Decision Diagram).

• Note: Further research and projects can be done using other edge detectors and compressors.
Conducted experiments

- I used a database which included images of different types: simple sketches, cartoons, individual objects and complex photos.
- Most of the comparisons made a good match of similar edge maps.
- Complex pictures which had a lot of noise that affected the edge detections had more difficulty to properly find a match.
Some Examples (Smallest NCD)

- Simple and similar contours of the same shifted object
Some Examples (Smallest NCD)

- Complex contours of shifted complex objects
Some Examples (Largest NCD)

- Simple and different contours of different objects
Conclusions

- The method works good with edge maps without a lot of noise, and therefore is efficient only for certain image databases.
- Some possibilities exist to improve the results of this method by using a different compressor and/or a different edge detection method.
- NCD is a good and efficient method of measuring distance between objects/information.
References

• J. Kiefer, P. Flajolet, and E-h. Yang. Universal Lossless Data Compression Via Binary Decision Diagrams
• Ming Li, Xin Chen, Xin Li, Bin Ma, and Paul M. B. Vitányi. The Similarity Metric
• Rudi Cilibrasi, Paul Vitányi and Ronald de Wolf. Algorithmic Clustering of Music Based on String Compression
• Xin-Jing Wang, Lei Zhang, Feng Jing and Wei-Ying Ma. Image Annotation Using Search and Mining Technologies