

LAB PIV

Òscar Medrano Sánchez

Abril 2024

Recuperació d'Imatges Basada en Continguts Descriptors basats en histograma

1 Introducció

El principal objectiu d'aquest informe és presentar els resultats d'una tasca pràctica destinada al desenvolupament d'un sistema de Recuperació d'Imatges Basada en Continguts (CBIR). El sistema està dissenyat per operar en una base de dades que conté 2.000 imatges diferents, agrupades en grups de 4 del mateix objecte amb diferent angle. L'objectiu és recuperar aquestes 4 imatges que corresponen al mateix objecte que la imatge d'entrada proporcionada al sistema.

La tasca pràctica va consistir en desenvolupar un sistema CBIR en MATLAB, utilitzant l'histograma de les imatges. El sistema es va provar en un conjunt d'imatges de consulta, i les imatges recuperades es van comparar amb la sortida esperada per avaluar el sistema.

2 Descripció global del sistema

El sistema pren un fitxer d'entrada, "input.txt", que conté els noms de les imatges a analitzar. Per a cada imatge especificada en el fitxer d'entrada, el sistema fa el següent:

Carrega la matriu H i extreu l'histograma h de la imatge actual. Calcula el vector de distància d entre h i tots els histogrames en H . A continuació, troba els 10 valors més petits al vector d i emmagatzema els seus índexs respectivament per poder escriure posteriorment els noms de les 10 imatges al fitxer de sortida, "output.txt".

En general, l'algoritme realitza una cerca de similitud per a cada imatge en el fitxer d'entrada, comparant el seu histograma amb tots els histogrames del conjunt de dades i seleccionant les 10 imatges amb les distàncies més petites. Les imatges seleccionades s'escriuen després al fitxer de sortida.

Algorithm 1: System

Input: Input file *input.txt*
Output: Output file *output.txt*
load('dataBase');
for $i \leftarrow 1$ **to** $NumImages$ **do**
 $h1 = \text{imhist}(\text{readImage}(im));$
 for $j \leftarrow 1$ **to** 2000 **do**
 $d(j) = \text{distanceX}(h1, H(j,:));$
 sort(d);
 for $j \leftarrow 1$ **to** 10 **do**
 write(Output, d(j));
return Output

Aquest sistema ofereix bastant joc perquè allà on hi ha *distanceX()* podem triar la forma de calcular distància que es vulgui provar en cada moment. Això permet provar varies distàncies per poder comprovar quina funciona millor i quina s'adapta més a la tasca.

Hem provat les següents 4 distàncies: distància de Bhattacharyya, de correlació, de Hellinger i Txi-quadrat.

Bhattacharyya

La distància de Bhattacharyya està definida com:

$$d(h1, h2) = \log_{10} \left(\sum_{n=0}^{N-1} \sqrt{h1 * h2} \right) \quad (1)$$

Sent $h1$ i $h2$ els histogrames que li entrem per funció, i així serà a totes les definicions de distàncies.

Correlació

La distància de correlació està definida com:

$$d(h1, h2) = 1 - \frac{\sum_{n=0}^{N-1} (h1 - \mu1)(h2 - \mu2)}{\sum_{n=0}^{N-1} (h1 - \mu1)^2 * \sum_{n=0}^{N-1} (h2 - \mu2)^2} \quad (2)$$

Hellinger

La distància de Hellinger està definida com:

$$d(h1, h2) = \sqrt{1 - \sum_{n=0}^{N-1} (\sqrt{h1 * h2})}; \quad (3)$$

Sent $h1$ i $h2$ la suma de totes les components dels histogrames que li entrem per funció, en aquest cas.

Txi-quadrat

La distància de Hellinger està definida com:

$$d(h1, h2) = \frac{\sum_{n=0}^{N-1} (h1 - h2)^2}{h1} \quad (4)$$

3 Resultats

Per comprovar quina de les 4 distàncies és la idònia visualitzarem els resultats mitjançant una gràfica de Recall-Precision:

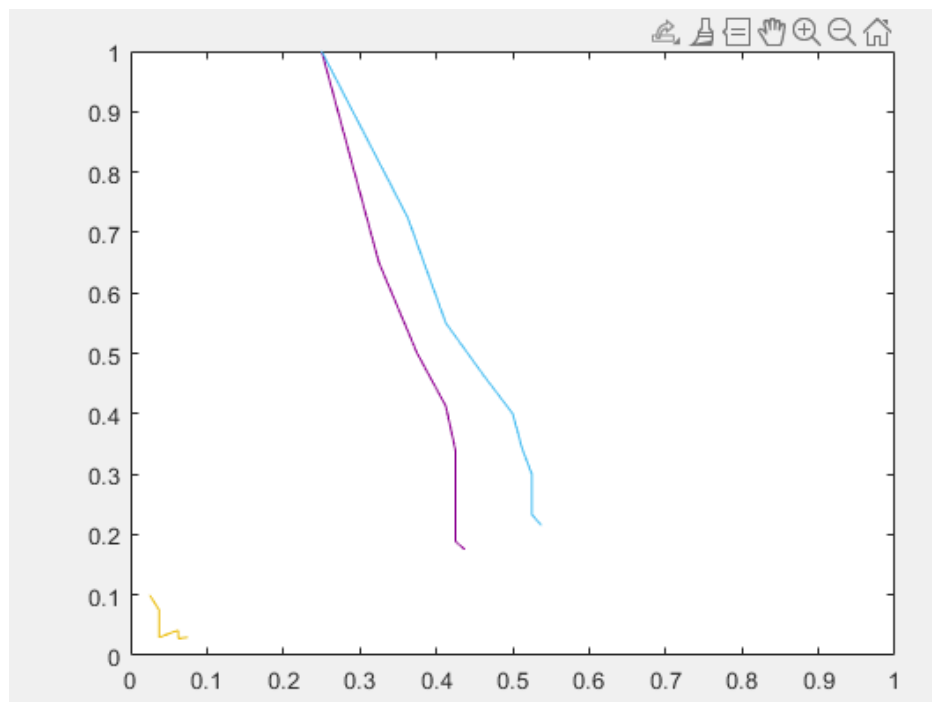


Figure 1: Recall-Precision Graph

L'eix horizontal representa el Recall i el vertical la Precisió, per tant, com més aprop estigui del vèrtex superior esquerre, més F-score té i per conseqüència més fiable és la distància. Veiem només 3 línies degut a que una està superposada a una altra. Hellinger i Bhattacharyya tenen exactament la mateixa gràfica, que és la de color blau, per tant són les distàncies que millor aproximen les diferències entre histogrames. Finalment, les altres dues, la lila i la groga que són la distància de correlació i la distància Txi-quadrat, i veiem que no funcionen tan bé com les anteriors.

En quant a cost computacional, aquesta captura ens mostra dos temps, el primer és el temps total que tarda en executar-se el programa, el segon és aquest dividit entre el número d'imatges que compara, és a dir, 20.

```
>> sessio2
Elapsed time is 0.266249 seconds.
>> sessio2

ans =

    0.0132
```

Figure 2: Temps

Aquest temps és el calculat amb la distància de Bhattacharyya. Com veiem, tarda al voltant de 13 ms per trobar les 10 imatges que tenen l'histograma més semblant.

4 Conclusions

Val la pena assenyalar que els resultats presentats aquí es basen en un experiment específic, concretament les imatges del fitxer input.txt, i podrien no ser generalitzables a altres contextos. Per tant, els resultats s'haurien d'interpretar amb precaució i podria ser necessari realitzar més experiments amb altres fitxers input.txt per confirmar les conclusions.

La configuració òptima per al nostre sistema sense tenir en compte el temps d'execució per a aquest fitxer d'entrada específic sembla ser utilitzar la distància de Battachary. No obstant això, creiem que per a altres fitxers input.txt, el rendiment podria no ser tan alt. D'alguna manera, estaríem ajustant excessivament el sistema per a sempre treballar amb les mateixes dades d'entrada. D'altra banda, tenint en compte que aquesta distància no és infalible, podríem optar per la distància de Hellinger, ja que té resultats exactament iguals que la distància de Battachary. Finalment, l'opció de treballar amb la correlació o la Txi-quadrat quedaria totalment descartada, degut al seu baix rendiment en Precisió i Recall. En conclusió, la configuració òptima per al sistema es basa en un experiment específic i pot no ser generalitzable a altres contextos. Podria ser necessari realitzar més experiments amb altres fitxers input per confirmar les conclusions.

Millores

Per millorar la cerca d'imatges similars en una base de dades utilitzant histogrames, hi ha diversos aspectes que podem considerar per aconseguir resultats més precisos i variats.

Utilitzar Histogrames de Color: En lloc de limitar-nos als histogrames de grisos, podríem incloure histogrames de color. Això ens permetria capturar millor les característiques cromàtiques de les imatges.

Utilitzar Múltiples Característiques: A més dels histogrames de color, podríem considerar l'ús d'altres característiques com textures, formes o descriptors de punts clau. Integrar aquestes característiques addicionals ens permetria capturar una àmplia gamma d'informació i millorar la robustesa del nostre sistema de cerca.