CENG 483

Introduction to Computer Vision

Fall 2021-2022

Take Home Exam 1

Instance Recognition with Color Histograms

Student ID: 2304582

Please fill in the sections below only with the requested information. If you have additional things you want to mention, you can use the last section. For all of the configurations make sure that your quantization interval is divisible by 256 in order to obtain equal bins.

1 3D Color Histogram

In this section, give your results without dividing the images into grids. Your histogram must have at most 4096 bins. E.g. Assume that you choose 16 for quantization interval then you will have 16 bins for each channel and 4096 bins for your 3D color histogram.

- Pick 4 different quantization intervals and give your top-1 accuracy results for each of them on every query dataset.
- Explain the differences in results and possible causes of them if there are any.

I have performed experiments with 2,4,8,16 bins for all three queries. Note that those are single channel bin numbers. For example for 4 bins, there are 4*4*4 = 64 bins in total.

Query 1 In Query1, images are exactly the same with the support images, jsut a little bit zoomed in. Top 1 accuracies are calculated as follows;

1. 2 bins : 0.93
2. 4 bins : 1
3. 8 bins : 1
4. 16 bins : 1

Query 2 In Query2, there are rotated versions of the support images. Top 1 accuracies are calculated as follows;

1. 2 bins : 1
2. 4 bins : 1
3. 8 bins : 1
4. 16 bins : 1

Query 3 In Query3, there are different color versions of the support images. Colors of the images are changed. Top 1 accuracies are calculated as follows;

1. 2 bins : 0.08
2. 4 bins : 0.115
3. 8 bins : 0.105
4. 16 bins : 0.11

Explain the differences in results and possible causes of them if there are any.

Since in the first query, most of the colors match with the support images, images match in a high rate. However, since there are differences in colors, histograms may differ as well. Yet, we are trying to find the one with minimum divergence, therefore we get a high success rate. They are zoomed versions of the original images where most of the pixels are the same. Therefore, except for the very discrete case (2 bins) we get a success rate of 200/200. 2 bins case is an exception where all the colors are stucked only at 2 possible locations.

In the second query, images are only the rotated versions of support images. That does not change the color histogram of an image. Therefore, we get a success rate of 100. Therefore, we get a success rate of 100 regardless of the number of bins.

In the third query, colors are played. Since changing the color changes the color histogram as well, and we are only taking the average, we are not good at finding the images. We do not care about shapes at all; thus, it is normal to fail when colors are changed. Even though we increase the bin number, we could not get a good result since colors are obviously different from the original images. For this query to match in high rate, another method that includes edge detection is necessary.

2 Per Channel Color histogram

In this section, give your results without dividing the images into grids.

- Pick 5 different quantization intervals and give your top-1 accuracy results for each of them on every query dataset.

  I have performed experiments with 2,4,8,16,32,64,128 and 256 bins for all three queries.

  Query 1 In Query1, images are exactly the same with the support images, just a little bit zoomed in. Top 1 accuracies are calculated as follows;

  1. 2 bins : 0.6
  2. 4 bins : 0.935
  3. 8 bins : 0.975
  4. 16 bins : 0.98
  5. 32 bins : 0.98
  6. 64 bins : 0.98
  7. 128 bins : 0.98
  8. 256 bins : 0.985

  Query 2 In Query2, there are rotated versions of the support images. Top 1 accuracies are calculated as follows;

  1. 2 bins : 1
  2. 4 bins : 1
  3. 8 bins : 1

4. 16 bins : 1
5. 32 bins : 1
6. 64 bins : 1
7. 128 bins : 1
8. 256 bins : 1

Query 3 In Query3, there are different color versions of the support images. Colors of the images are changed. Top 1 accuracies are calculated as follows;

1. 2 bins : 0.04
2. 4 bins : 0.14
3. 8 bins : 0.135
4. 16 bins : 0.125
5. 32 bins : 0.125
6. 64 bins : 0.105
7. 128 bins : 0.1
8. 256 bins : 0.1

Since in the first query, most of the colors match with the support images, they match in a high rate. However, since there are differences in colors, histograms may differ as well. Yet, we are trying to find the one with minimum divergence, therefore we get a high success rate. They are zoomed versions of the original images where most of the pixels are the same. We get a worse result than three channel version. We keep getting better results as we increase the detail level in this query as can be seen from the test results. The reason for 3 channel histogram to be more successful is that it is more detailed. In conclusion, Query 1 images match in a very high rate. It gets better as we increase the bin number, reducing the bin interval, which means increasing the detail level.

In the second query, images are only the rotated versions of support images. That does not change the color histogram of an image. Therefore, we get a success rate of 1 regardless of the number of bins. In query 2, the quantization used does not change the result since no matter what image has exactly the same histogram with the support image.

In the third query, colors are played. Since changing the color changes the color histogram as well, and we are only taking the average, we are not good at finding the images. We do not care about shapes at all; thus, it is normal to fail when colors are changed. Even though we increase the bin number, we could not get a good result since colors are obviously different from the original images. Success rate as a little bit higher than the 3 channel case since we are taking average here, and some images which are only changed in one color value approximates to the original one better. However, it is far from perfect.

Before starting the next section, please pick up the best configuration for two properties above and continue with them.

3 Grid Based Feature Extraction - Query set 1

Give your top-1 accuracy for all of the configurations below.

I have chosen to work with 4 bins for 3d channel histogram and 16 bins for the 1 channel histogram.

3.1 48×48 spatial grid

- 3d color histogram: 1
- per-channel histogram: 1

3.2 24×24 spatial grid

- 3d color histogram: 0.98
- per-channel histogram: 1

3.3 16×16 spatial grid

- 3d color histogram: 0.9
- per-channel histogram: 1

3.4 12×12 spatial grid

- 3d color histogram: 0.775
- per-channel histogram: 0.98

3.5 Questions

- What do you think about the cause of the difference between the results?
- Explain the advantages/disadvantages of using grids in both types of histograms if there are any.

Accuracy is good in general. Since in query 1, images are very similar to the support images, we get perfect results with per channel histogram up to some point. At some point, our intervals are too detailed so that our algorithm starts to make mistakes. In the 3D case, we get worse results even though they are good in general. That results from the fact that in 3d color histogram, quantization intervals contain very specific colors. Therefore, with too many bins involved, we start to lose our accuracy.

In specific, per channel accuracy is increased. That results from the fact that when we divide the image into pieces, even though there are different RGB values in some pixels, since divided grids match, images match in a high accuracy. However, when we increase the detail level too much as in the case 12x12 grid, we start to lose some of the images.

3 channel results were very accurate without using grid. After some detail level, we obtained perfect results, 1 accuracy. We see here that it is not the case and algorithm cannot match some of the images correctly. The reason for that is that if we divide the image into many grids and also have a three channel bin, it is way too detailed than it should be. Therefore, since the images are not exactly the same, the algorithm produces divergences.

The most critical question is why? The reason for this is that pixel that have the same values do not have exact coordinates since we zoomed the images. That is why, as we keep decreasing the grid size we do have high divergence values since pixels with the same RGB values may not be located into the same grid. That results in even though in the general sense, two images have many pixels in common, since they are not compared against each other, we lose our high accuracy values.

Increasing the bin number for the query 1 is redundant and even bad at some point. That is why if we have to, we should use only for per channel histograms without decreasing the grids too much such as 48x48 or 24x24.

4 Grid Based Feature Extraction - Query set 2

Give your top-1 accuracy for all of the configurations below. I have chosen to work with 8 bins for 3d channel histogram and 8 bins for the 1 channel histogram.

4.1 48×48 spatial grid

- 3d color histogram: 0.525
- per-channel histogram: 0.355

4.2 24×24 spatial grid

- 3d color histogram: 0.285
- per-channel histogram: 0.14

4.3 16×16 spatial grid

- 3d color histogram: 0.2
- per-channel histogram: 0.1

4.4 12×12 spatial grid

- 3d color histogram: 0.2
- per-channel histogram: 0.08

4.5 Questions

- What do you think about the cause of the difference between the results?
- Explain the advantages/disadvantages of using grids in both types of histograms if there are any.

First of all, since in the previous part, we get a perfect accuracy for all the results, I have chosen the bin numbers by trying which gives the best results. Dividing the image into grids has a very negative effect on query 2. The reason is that query 2 is the rotated versions of the support images. When we divide the image into grids, divided parts of the image do not match with each other. Some part of the equivalent pixels of that image stays in another grid. Therefore for that sub image we get a high divergence value and our accuracy falls dramatically. As we keep dividing the image into more grids, our chance to have the very same parts of the images in the same grid gets smaller and smaller. Therefore, our accuracy falls even more. If we could have 9216 grids meaning that each pixel is another grid, our program would give random results since we do not expect the colors of those pixels exactly. In a rotated image, as dividing in into subparts, we are decreasing the probability that the very same part of the images are compared against each other.

In 3d case, results are slightly better since at least it considers all the combinations of the colors. In this way, if the RGB values of the neighbour pixels are close, for slightly rotated images, there may be matches. However, using grid for query 2 is meaningless.

5 Grid Based Feature Extraction - Query set 3

Give your top-1 accuracy for all of the configurations below. I have chosen to work with 4 bins for 3d channel histogram and 8 bins for the 1 channel histogram.

5.1 48×48 spatial grid

- 3d color histogram: 0.13
- per-channel histogram: 0.22

5.2 24×24 spatial grid

- 3d color histogram: 0.14
- per-channel histogram: 0.24

5.3 16×16 spatial grid

- 3d color histogram: 0.12
- per-channel histogram: 0.25

5.4 12×12 spatial grid

- 3d color histogram: 0.125
- per-channel histogram: 0.24

5.5 Questions

- What do you think about the cause of the difference between the results?
- Explain the advantages/disadvantages of using grids in both types of histograms if there are any.

  In 3d color histogram case results did not change. Considering the computational costs of grids, it is redundant to have such an implementation. However, in one channel color histogram approach, using grids doubled the accuracy. The reason for that might be the similarity of some color channels between two images. For example in an image in Query 3, there might a small region where R and G colors match with the original image. We calculate a low divergence value for those two channels and take average. In 3 channel case, we could not be able to match that small grid since they go to the different bins. And grids help us to capture those small areas. As we divide the image into grids, it is most likely that small areas that match in some channels are dominant in the grid and query image matches with the original image. However, still we are using a color based histogram, thus, our accuracy is at most 0.25. Disadvantage of this method is that, complexity increases a lot as we increase the number of bins and decrease the grid size.

6 Additional Comments and References

I did not use any outside source.