CENG 483

Introduction to Computer Vision

Fall 2022-2023

Take Home Exam 1

Instance Recognition with Color Histograms

Student ID: 2010023

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.

Glossary:

- S: Support set, containing raw images.
- $Q1$: Query-1 set, containing the cropped images from set S of different scales. These images can be viewed as zoomed-in crops of S.
- $Q2$: Query-2 set, containing the raw images with different rotation transformations.
- $Q3$: Query-3 set, containing the raw images with different color formations.
- d: Embedding size of the color histogram, i.e d dimensions or number of bins.
- Top-1 Accuracy: Metrics throughout the paper is given and discussed as a percent [0-100] to ease reading, i.e. 95.43 means %95.43.
- Discussion: To ease reading over paper, we introduced *Discussion* subsection under the sections, and conducted the discussion and answers related to that topic/section under this subsection.

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.

d	bin-size	Query-1	Query-2	Query-3	Avg. Performance
2^3	128	94.00	100.00	7.00	67.00
4^3	64	100.00	100.00	**14.50**	**71.50**
8^3	32	100.00	100.00	12.00	70.67
16^3	16	100.00	100.00	13.00	71.00

Table 1: Top-1 Accuracy metrics for different bin sizes for 3D color histogram.

1.1 Discussion

The results for 3D color histogram with 4 different bin sizes is given in Table 1. All bin sizes are equally and almost equally perform for $Q2$ and $Q1$ respectively.

- Results for $Q1$ is similar and high for different bin sizes, this is expected because the (possibly center or close-center) crop of scaled image still keeps the explanatory color distribution information.
- We have equally performing bin sizes for $Q2$, this is expected because a color histogram embedding is invariant the rotation transformation, i.e rotation transformation shifts the pixel values (in a systematic way) which does not change the color distribution at all. This has an obvious drawback that if the pixels of an image from S were randomly shuffled we would've got the same metrics, but the image would be looking as a noise rather than a bird (this is irrelevant for our query set $Q2$).
- $Q3$ is the hardest query set to match images as the query images' color distribution are different than the gold images (S) as it is illustrated in Figure 1. $d = 64$, i.e 4^3 number of bins, yields the best results on $Q3$, but there is no pattern such that increasing d yields better metrics. $d = 8$ (2^3) is too small and there is too much information lost in embedding, so it is not unexpected that it performs poorly. $d = 8^3$ and $d = 16^3$; however, cannot reach the performance of $d = 64$. One reason for this might be that it becomes harder to match the joint distribution as d increases. In other words, embedding becomes more sensitive to small changes as d increases, and the increase in 3d histogram embedding size is not linear, but cubic instead w.r.t per-channel bin size.

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

d	bin-size	Query-1	Query-2	Query-3	Avg. Performance
16	16	96.50	100.00	14.00	70.17
32	8	96.50	100.00	**14.50**	**70.34**
64	4	**97.00**	100.00	14.00	**70.34**
128	2	96.50	100.00	13.50	70.00
256	1	**97.00**	100.00	13.00	70.00

Table 2: Top-1 Accuracy metrics are given for different number of bins for per-channel color histogram. Dimension d in the table refers to size for a single channel, and thus for an image the resulting $d' = (3, d)$ due to channel size (3).

2.1 Discussion

The results for per-channel color histogram with 5 different bin sizes is given in Table 2. All bin sizes are equally and almost equally perform for $Q2$ and $Q1$ respectively, similar to 3D color histogram. The explanation of this table is identical to the ones in Section 1.1.

Configuration

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

The chosen configurations are $d = 64$ (bin-size = 64) for 3D Histogram and $d = 64$ (bin-size = 4) for Per-Channel histogram. Although there are competing configurations for per-channel histogram ($d = 32$ and $d = 64$), we chose $d = 64$ as it kinda both favors Query-1 and Query-3 set, and also contains more information compared to $d = 32$.

3 Grid Based Feature Extraction - Query set 1

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

3.1 2×2 spatial grid

- 3d color histogram: 100.00
- per-channel histogram: 99.50

3.2 4×4 spatial grid

- 3d color histogram: 100.00
- per-channel histogram: 100.00

3.3 6×6 spatial grid

- 3d color histogram: 100.00
- per-channel histogram: 100.00

3.4 8×8 spatial grid

- 3d color histogram: 100.00
- per-channel histogram: 100.00

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.

3.6 Discussion

There are no differenece in the results for different grid size for both 3d and slight difference per-channel histograms only in 2×2 grid. 3D histogram with $d = 64$ with $grid = (1, 1)$ has performance metric 100.00 for $Q1$, and applying grids did not decrease the performance.

Per-channel histogram with $d = 64$ with $grid = (1, 1)$ has performance metric 97.00 for $Q1$, and applying grids has yielded increase in top-1 accuracy. This makes sense because increasing grid size n gives n^2 grid cells and since $Q1$ set contains (centered-almost centered) "zoomed-in" images of S, so the more grid cells close to the center over image, the better performance we have. The reason is that center based grid cells would be more frequent and more dominantly affect the average operation.

4 Grid Based Feature Extraction - Query set 2

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

4.1 2×2 spatial grid

- 3d color histogram: 57.00
- per-channel histogram: 44.00

4.2 4×4 spatial grid

- 3d color histogram: 46.50
- per-channel histogram: 34.00

4.3 6×6 spatial grid

- 3d color histogram: 40.00
- per-channel histogram: 29.50

4.4 8×8 spatial grid

- 3d color histogram: 39.00
- per-channel histogram: 25.00

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.

4.6 Discussion

There are drastic changes in the metrics for both 3D and per-channel histograms. This is expected because $Q2$ includes rotated images from S. Although we stated that color histograms are invariant to rotation, with grids we introduce local regions (grid cells) and compute color histograms from those regions. Since the images are rotated, the corresponding grid cells does not necessarily have to contain similar color distributions for that region and the color distribution we have for ground truth image (from S) is now lost with the grids due to rotation. However, we can think that if the instance for example is in the center of the image and the background is similar for all directions from center, then we can anticipate that we probably still have a match. That's probably the reason that top-1 accuracy does not drop to 0 immediately.

Another observation is that if we increase the grid size, the top-1 accuracy tends to decrease as well, this is because with more grid cells we make the computations more sensitive to the local changes as also stated in Proposition 1. The increase in grid cells is quadratic in this square grid form, so we increased from $2 \times 2 = 4$ towards $8 \times 8 = 64$ in which we increase the number of local regions in a quadratic order.

In addition to these, we can make an additional observation that 3d color histogram performs better than per-channel histogram regardless of the grid size. For this we can take the statement of Proposition 2 into account. The possible reasoning behind this is that with 3d color histogram it is harder to match a query image to an image with different context (e.g. background, instance color, etc.).

5 Grid Based Feature Extraction - Query set 3

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

5.1 2×2 spatial grid

- 3d color histogram: 19.00
- per-channel histogram: 24.00

5.2 4×4 spatial grid

- 3d color histogram: 32.00
- per-channel histogram: 37.00

5.3 6×6 spatial grid

- 3d color histogram: 38.00
- per-channel histogram: 42.00

5.4 8×8 spatial grid

- 3d color histogram: 42.00
- per-channel histogram: 45.50

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.

5.6 Discussion

There are drastic changes in the metrics for both 3D and per-channel histograms similar to Section 4. However, the change is different in terms of performance as the performance increased with the increase in grid size.

The first observation is that increase in grid size results in increase in performance. This expected because $Q3$ contains color shifted images from S, the changes are mostly (not completely) local (e.g. color of the bird is different, but the background is similar). Thus, dividing into more grids helps us mitigating the performance loss due to instance change in a local region (see Proposition 1) and affecting the whole image's color distribution. For example, take a look at Figure 1, where we see some local changes in the R channel as pixels at bottom left have higher intensity in the ground truth, also the instance (the bird) has more high intensity pixels. However, when we look as a whole, overall the low intensity regions are mostly similar (i.e. background), and thus, say 2×2 grid separates bottom left as a individual region where the similarity is low, but other three grid cells helps matching the embedding vectors as they are more similar.

The second observation is that per-channel histogram performs better than 3d histogram regardless of the observed grid size contrarily to Subsection 4.6. The possible reason for this is again Proposition 2, as opposed to $Q2$, $Q3$ contains images with color shifts, so we cannot make the same point, rotated image still contains color distribution in a sense, anymore that we mentioned in Subsection 4.6. This color shifts makes joint distribution harder to match with the same original image. Per-channel performs better probably due to the proposition's statement as per-channel is less sensitive to inter-channel changes.

6 Additional Comments and References

6.1 Example 1

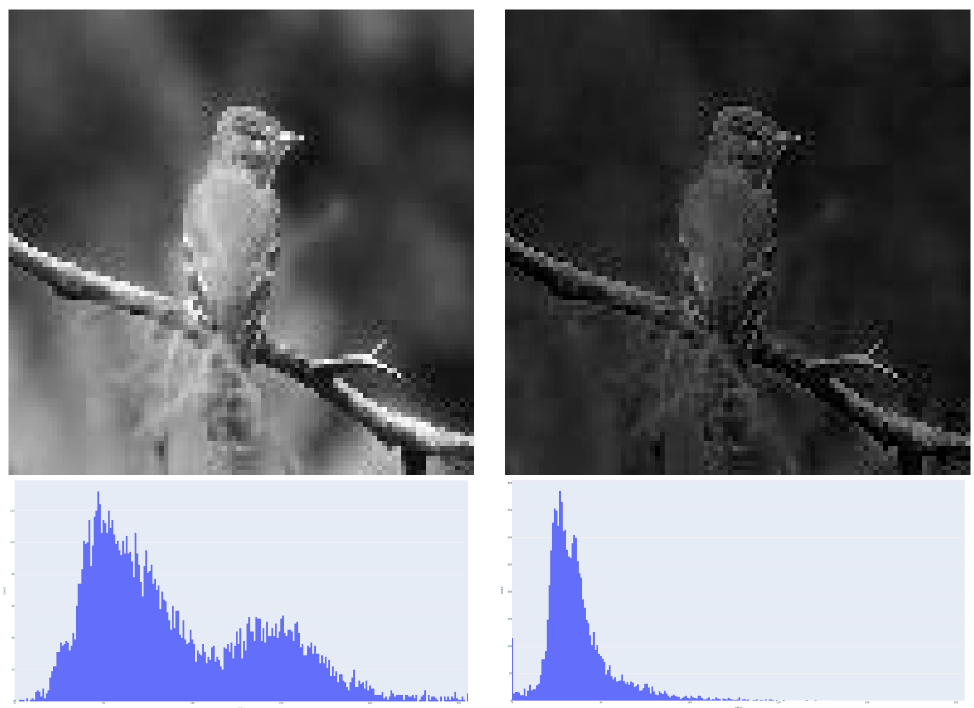

Figure 1: Visualizations of *Acadian_Flycatcher_0016_887710060.jpg* (only for channel R as grayscale) and their color histograms below each. The one on the left is from S, and the image on the right is from $Q3$.

6.2 Propositions

Proposition 1. *Increasing grid size makes the embedding vectors and consequent similarity/dissimilarity computations more sensitive to local changes.*

Thought experiment: *What if we make grid size large as much as we can ? We would get 1 pixel per grid cell, that is grid size is the image size itself, then we can immediately relate what would the result be if we compare pixel by pixel similarity/dissimilarity on each grid and average them for a rotated image.*

Proposition 2. *3D color histogram is more sensitive to inter-channel (across channels) changes compared to per-channel histogram depending on the embedding size d. In other words, slight changes in a single channel yields more dramatic changes in joint distribution rather than per-channel distributions.*

Thought experiment: *Let's say that we want to assess the similarity between two RGB images of size (w,h). Let the first image be a blue image where* $(x_i, y_i) = (0, 0, 255)\ \forall\ i$ *and the second image be a green image* $(x_i, y_i) = (0, 255, 0)\ \forall\ i$*. If we look at individual histograms per channel, we will see that channels G and B are not similar, but channel R for both images are exactly the same, so we have some similarity depending on our metric. However, if we take a look at joint distribution, they are completely different as there are no same/similar values occuring for all RGB channel.*