Counting jelly beans: voxel carving and segmentation of container of heterogenous objects

Alex Cope, Mark O'Meara

March 2014

Abstract

Foo

1 Introduction

2 Prior Works

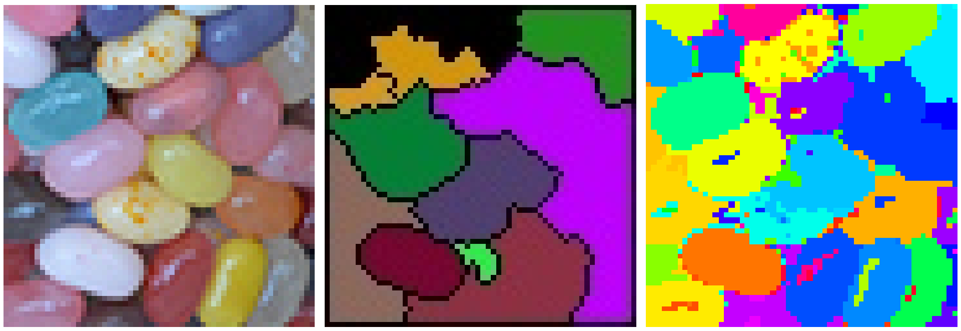
I can talk about this: counting moving objects, generally.

3 Approach

3.1 Voxel Carving

3.2 Segmentation

We tried multiple approaches for solving the segmentation problem. The difficulty with using segmentation to count the number of objects in an image is that the algorithm must be robust to both false positives and false negatives; the count must be exact. The greater the number of objects in the scene, the more difficult this becomes.



(a) Unaltered. (b) Watershed. (c) Mean-shift.

Figure 1: Unaltered photo (a). Watershed (b) and mean shift (c) segmentation results. Different colors represent different clusters. (The black lines in the watershed picture are for clarity; they are omitted in the mean-shift picture due to the number of small clusters.)

3.2.1 Watershed

We tried a couple classic blob detectors to count the number of beans and found limited success. The watershed algorithm, developed by Lindeberg (1993), detects blobs using local extrema in the image space [**?**]. The

3.2.2 Mean-shift

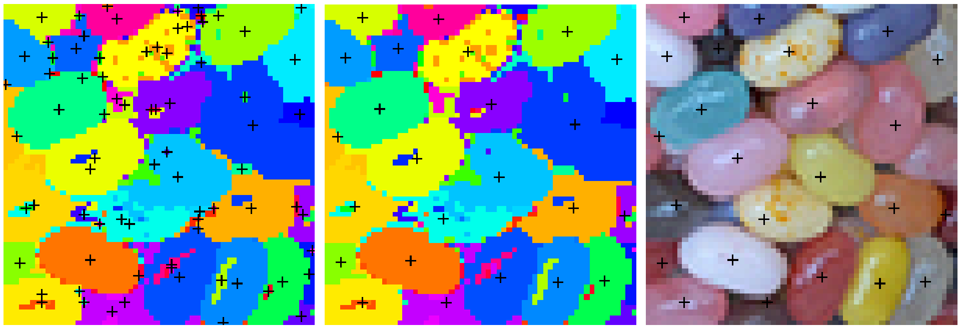
4 Experiments

4.1 Segmentation

Different segmentation approaches were testing on the same small (60x60) cropped image of jellybeans (Fig. 1a). A cropped image was used to speed up runtime while iterating through refinements; this was necessary for mean-shift, which runs in $O(n^2)$ and has two user defined parameters that drastically affect the final result.

Watershed (Fig. 1b) performs poorly compared to mean-shift; it is able to correctly identify one jellybean (in maroon) and otherwise grossly underestimates the number of clusters.

Mean-shift (Fig. 1c) performs much better. However, we can see from the image that it is susceptible to noise and texture (for example, see the



(a) All clusters. (b) Filtered clusters. (c) Over image.

Figure 2: Cluster centroids found using mean-shift.

number of small clusters it detects on the speckled jellybean at the top) and thus overestimates the number of clusters.

However, we can mitigate this overestimate by culling the clusters using a simple heuristic. We calculate the centroid, area, and variance of each cluster detected by the mean-shift algorithm. If the area is below and the variance above user-set thresholds, then the cluster is ignored. We see the results in Fig. 2. Fig. 2a shows all of the cluster centroids detected by mean-shift ($N = 78$), and Fig. 2b shows only the clusters with area > 25 and variance < 1000 ($N = 24$). These threshold values are obviously affected by the scale of the image; however, because the noisy clusters are all small and all of the jellybeans are roughly the same size, the threshold values can be estimated roughly (i.e. they do not have to be fine-tuned for good performance.)

Quantitatively, we see that this cluster detection algorithm performs remarkably well (Fig. 2c). It is able to correctly count all but one jellybean on the front layer, and even correctly identifies some of the occluded jellybeans below the front layer.

5 Conclusion

References

[1] Serge Beucher, *Image segmentation and mathematical morphology.* http://cmm.ensmp.fr/ beucher/wtshed.html, May 2010.