

Image Analysis Coursework

James Hughes

Word count: 0

June 7, 2024

Contents

1	Introduction	3
2	Image Segmentation	3
2.1	CT Image	3
2.2	Flowers Image	4
2.3	Coins Image	5
3	Exploiting Sparsity in Signal Processing	7
4	Algorithms for Solving Inverse Problems	7
4.1	Convergence of Gradient Descent	7
5	Discussion	7
A	Statement on the use of auto-generation tools	8

1 Introduction

2 Image Segmentation

2.1 CT Image

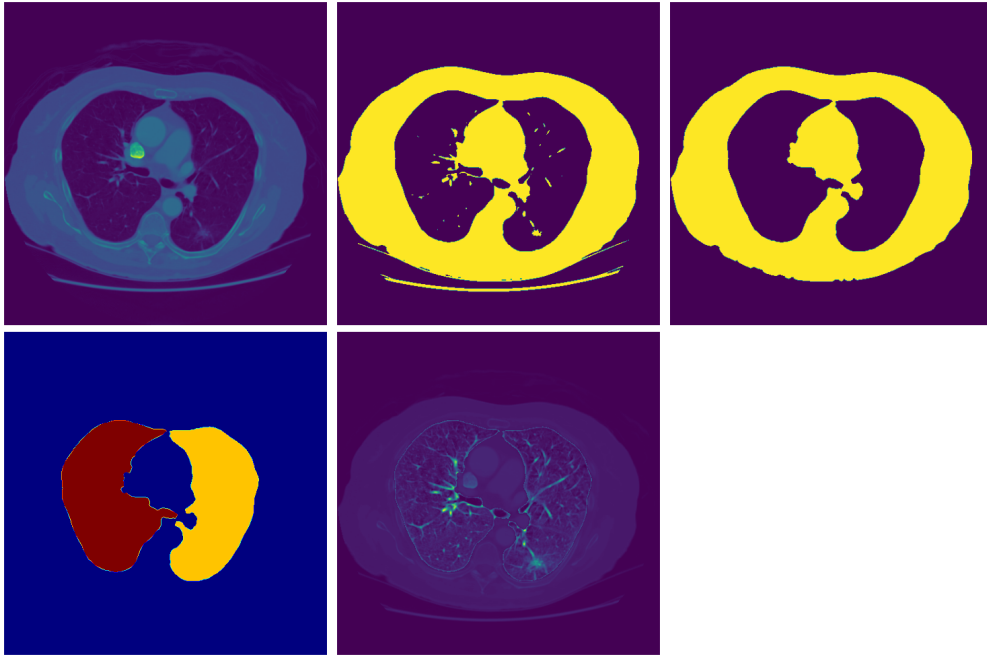


Figure 1: Steps of CT Segmentation

The CT scan image was segmented using Otsu's method for image segmentation. This was an appropriate, simple method which is suited to the monochrome image. This method easily segments the tissue and bone surrounding the lungs, distinguishing this from the lungs themselves and the surrounding background. The segmented image was then post-processed with 'binary closing', a morphological operation, as well as Scikit-Image's `morphology.remove_small_objects` method. The four segmented regions of the image were then labelled, and the lungs chosen as the smallest masks by area.

2.2 Flowers Image

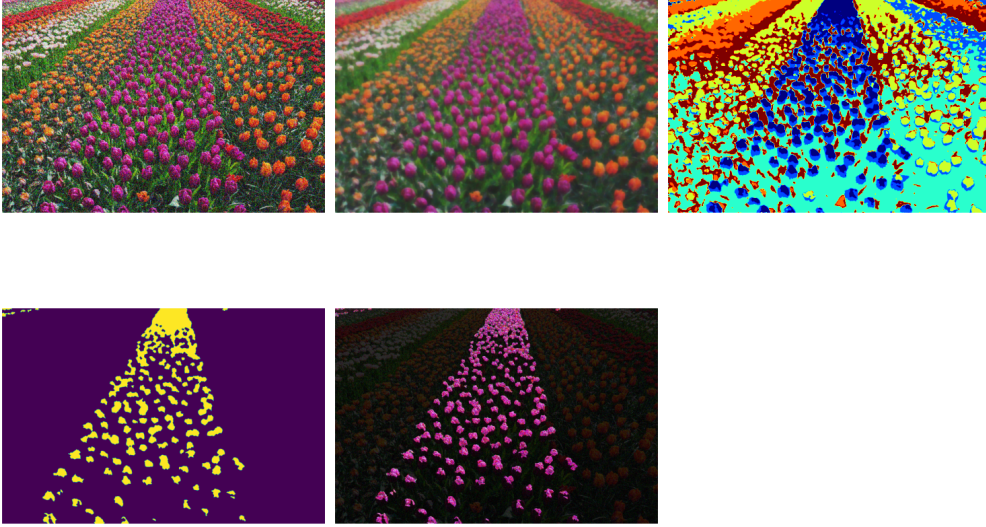


Figure 2: Steps of Flowers Segmentation

For the flowers image, a more sophisticated segmentation was required in order to handle the three colour channels of the pixels, as well as to produce a more complex segmentation mask composed of lots of small disconnected regions. Since the desired segmentation was to be based on the colours present in the image, it was less important to incorporate local pixel information into the image, therefore making clustering an appropriate strategy. In particular, KMeans was chosen, since this method was simple to implement, commonly used for image segmentation, and enabled a fixed number of clusters to be specified. In this case, six clusters was decided as a meaningful number, corresponding roughly to two shades of green (grass) and four flower colours. KMeans was implemented as a `class` in the `ml` module according to Lloyd's Algorithm [1], using the `kmeans++` [2] centroid initialisation method.

Based on this choice of algorithm, Total Variation denoising was used to pre-process the image. This meant that in the pre-processed image, pixel values were coerced to be similar to those nearby. This ensured that the

clustering algorithm was more stable (since it artificially made the cluster structure of the data more pronounced), and that the segmentation masks weren't too fine-grained.

The same morphological post-processing from the CT image was used again, in order to remove segmentation masks that were smaller than the flower buds.

2.3 Coins Image

A strong degree of preprocessing was helpful for this image, to assist the segmentation algorithm. Firstly, the image has been corrupted with vertical columns of pixels that have been masked with zero values. This was addressed by applying a local median filter. The locality of this filter was set to each pixel plus its 6 closest horizontal neighbours. This was wide enough to ensure that in most cases the majority of these 7 pixels are not corrupted, even in regions where the image is affected more strongly. An unsharp filter is then used to increase the difference in pixel values within the coin regions compared to just outside of these regions.

To segment all of the coins, the Chan-Vese [3] segmentation algorithm was used. This method relies on the iterative minimisation of an energy function, which describes the desired segmentation properties mathematically. In particular, this technique models the segmentation mask as the set $x \in \Omega : \phi(x) > 0$, where $\Omega \subset \mathbb{R}^2$ is the space representing the image, and $\phi : \Omega \rightarrow [0, 1]$ is a function which evolves over the course of the computation. The function is evolved iteratively to minimise the objective

$$F(\phi, c_1, c_2) = \mu \int_{\Omega} \delta(\phi) |\nabla \phi| dx dy + \nu \int_{\Omega} H(\phi) dx dy \\ + \lambda_1 \int_{\Omega} |u_0 - c_1|^2 H(\phi) dx dy + \lambda_2 \int_{\Omega} |u_0 - c_2|^2 (1 - H(\phi)) dx dy$$

where $\mu, \nu, \lambda_1, \lambda_2$ are all fixed hyperparameters. The first two terms serve to regularise the segmentation, by minimising the length of the segmentation boundary and the segmented area respectively (the area term is ignored in the code). The second two terms seek to minimise the deviations of pixel values within, and outside, the segmented area from two distinct 'representative

values', c_1 and c_2 . It is worth noting that for any ϕ , these values minimise the energy when they are the average pixel values in the two corresponding areas of the image.

The algorithm highlighted the background of the image. Thus to get masks of the coins, the segmentation values were permuted in 0,1 and the segmentation was again post-processed with a morphological closing operation. The masks were then labelled, and the correct labels were identified via 'test pixels', which are seen marking the original image (top-left) in yellow in Figure 3

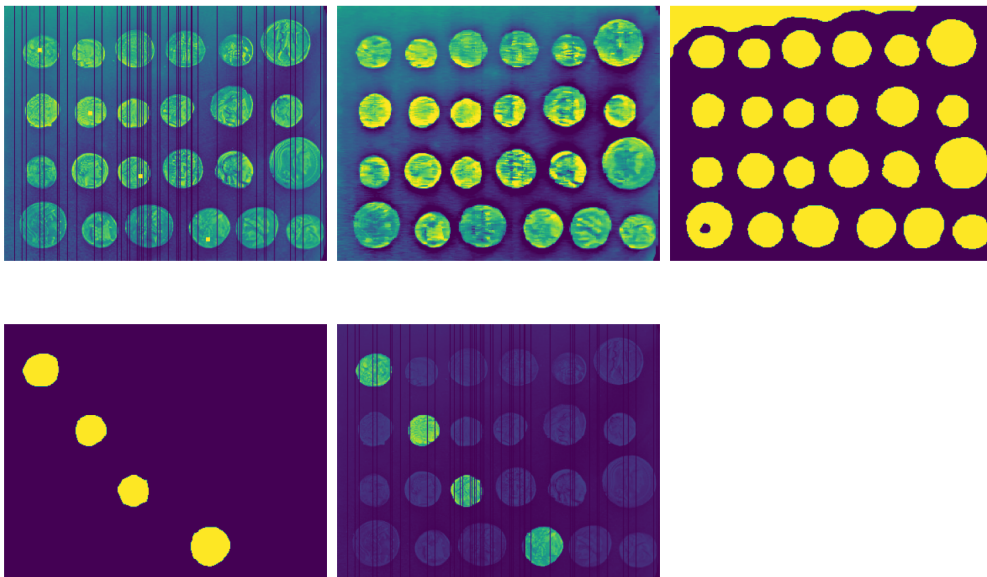


Figure 3: Steps of Coins Segmentation

3 Exploiting Sparsity in Signal Processing

4 Algorithms for Solving Inverse Problems

4.1 Convergence of Gradient Descent

Gradient Descent.

We can prove that the given function $f : \mathbb{R}^2 \rightarrow \mathbb{R}$ defined

$$f(x_1, x_2) = x_1^2 + \frac{x_2^2}{2}$$

is L -smooth with $L = 2$ via the following

The result given is, for learning rate $\eta = \frac{1}{L}$, and an L -smooth function f ,

$$f(x_K) - f(x^*) \leq \frac{L\|x_0 - x^*\|_2^2}{2K}$$

It is important to note that this is an estimate that gives the accuracy as $\mathcal{O}(\frac{1}{K})$. We can use it to compute the estimate the number of steps to required to reach $\epsilon = 0.01$, but this will be an upper bound. Nonetheless, we can set the right-hand side to ϵ and rearrange to give:

$$K = \frac{L\|x_0 - x^*\|_2^2}{2\epsilon}$$

Substituting $\epsilon = 0.01$, $x^* = (0, 0)$, $x_0 = (1, 1)$, $L = 2$, we get $K = 200$.

5 Discussion

Results/conclusions Further work What I learned How I could have improved
[5]

References

- [1] D. MacKay, *Information Theory, Inference and Learning Algorithms*. Cambridge University Press, 2003.

- [2] D. Arthur and S. Vassilvitskii, “k-means++: the advantages of careful seeding,” in *Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, ser. SODA '07. USA: Society for Industrial and Applied Mathematics, 2007, pp. 1027–1035.
- [3] T. Chan and L. Vese, “An active contour model without edges,” in *Scale-Space Theories in Computer Vision*, M. Nielsen, P. Johansen, O. F. Olsen, and J. Weickert, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 1999, pp. 141–151.
- [4] M. Kass, A. Witkin, and D. Terzopoulos, “Snakes: Active contour models,” *International Journal of Computer Vision*, vol. 1, 1988.
- [5] X. Li *et al.*, “Three-dimensional structured illumination microscopy with enhanced axial resolution,” *Nature Biotechnology*, vol. 41, pp. 1307–1319, 2023. [Online]. Available: <https://doi.org/10.1038/s41587-022-01651-1>

A Statement on the use of auto-generation tools