DECODING SATELLITE IMAGES OF EARTHQUAKES, PART II: FEATURE EXTRACTION FROM IMAGES, EDGE DETECTION

Saumik Dana^{1,}*

¹*University of Southern California, Los Angeles, CA 90089*

*Address all correspondence to: Saumik Dana, E-mail: sdana@usc.edu

This document is part II of a series of documents providing a lowdown on image processing in the context of understanding satellite images of earthquakes

KEY WORDS: *Earthquakes, imaging*

FIG. 1: Annualized Earthquake Losses (AEL: estimated long-term value of earthquake losses to the general building stock) by County. Sources: (1) <https://www.usgs.gov/news/usgs-collaborates-fema-national-earthquake-loss-estimate> and (2) <https://www.fema.gov/hazus>. The analysis yielded an estimate of the national AEL of \$6.1 billion per year. Loss estimates based on the best science and engineering available (during 2016-2017). The study only addresses direct economic losses to buildings, and does not cover damage and losses to critical facilities, transportation and utility lifelines or indirect economic losses.

Ground motion and failure due to earthquakes lead to huge losses to the tune of billions of dollars annually in the United States (see Fig. 1). Detecting and cataloging these earthquakes are key to understanding their causes (natural or human-induced (Dana (2018, 2019); Dana et al. (2018, 2020); Dana and Jha (2021); Dana and Wheeler (2018a,b, 2019); Dana et al. (2021); Foulger et al. (2018))) and, ultimately, to mitigating the seismic risk. There are in essence, two approaches

- ✓ Template matching i.e. signal processing, with the input being the seismogram signals. This approach has a historic precedence, and is most often used due to its ubiquity (Valentine and Trampert (2012); Yoon et al. (2015))
- ✓ Image processing i.e. Computer vision, with the input being the satellite images of earthquakes. This approach

has been taking shape this decade since the advent of GPU computable neural network arithmetic computations (Krizhevsky et al. (2012)), while leveraging all the progress in the realm of medical image processing (McCann et al. (2017))

Before delving any further into computer vision, it is critical to understand how an image is looked upon as in the realm. It is critical to note that all these algorithms and computations are already implemented in standard libraries, nevertheless subject matter knowledge (SMK) is indeed the first step in leveraging these libraries. **Leveraging computer code without SMK is like explaining gravity to a chicken.** In computer vision society, a feature is defined as a function of one or more measurements, each of which specifies some quantifiable property of an object, and is computed such that it quantifies some significant characteristics of the object. There are broadly two feature types

- ✓ low level features
 - Edge detection
 - Image segmentation
 - Texture
 - Corners
- ✓ high level features
 - Point pattern
 - Shape
 - Contour

0.1 Edge detection

FIG. 2: Comparison of Edge Detection Techniques on Lena Image Original Image (b) Canny Method (c) Roberts Edges (d) LOG edges (e) Sobel. Source: Maini and Aggarwal (2009)

Edges are paid much attention due to its computational cheapness, because differences in pixel values between regions are relatively easy to calculate by considering gradients (Maini and Aggarwal (2009)). More importantly, they provide strong visual clues that can help the recognition process. Most edge detection methods work on the assumption that an edge occurs where there is a discontinuity in the intensity (or depth) function or a very steep intensity (or depth) gradient in the image. All edge detection algorithms try to locate the edges in an image presented by true (real) scene elements but not noise. They can be classified into categories

- ✓ First order derivative
- ✓ Second order derivative
- ✓ Canny edge detector

0.1.1 First order derivative

In computer vision, edge detection is traditionally implemented by convolving the signal with some form of linear filter, usually a filter that approximates a first or second derivative operator. Example of simplest operator is

$$\begin{bmatrix} -1 & 0 & 1 \end{bmatrix} \text{ and } \begin{bmatrix} -1 \\ 0 \\ 1 \end{bmatrix}$$

0.1.2 Second order derivative

Examples of operators are: laplacian operator implemented digitally using the filters $\begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix}$ and $\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$ or LoG (Laplacian of the Gaussian) operator where the image is smoothed before using Laplacian operator

0.1.3 Canny edge detector

Algorithm 1: Canny edge detection algorithm

Smooth image; // typically gaussian
Compute gradients; // Compute the gradient of the smooth surface of the convolved image function in any direction from the known gradient in any two directions
Suppress non-maximums; // Each pixel in turn forms the center of a nine-pixel neighborhood. By interpolation of the surrounding discrete grid values, the gradient magnitudes are calculated at the neighborhood boundary in both directions perpendicular to the center pixel. If the pixel under consideration is not greater than these two values (i.e. non-maximum), it is suppressed.
Threshold; // An upper and lower edge value limits are set. If a value lies above the upper threshold limit it is immediately accepted. If the value lies below the low threshold it is immediately rejected. Points which lie between the two limits are accepted if they are connected to pixels which exhibit strong response.
Aggregate; // General principle is to start from the smallest scale, then synthesize the larger scale outputs. The synthetic data was produced by convolving a Gaussian of larger scale with the smaller scale edge data. The procedure is repeated to mark edges at the second scale that were not present at the first, then third to second and so on.

Edge detection is treated as a signal processing problem to design a "optimal" edge detector (Canny (1986)). A comparison of the edge detection techniques is provided in Fig. 2

References

- Canny, J., A computational approach to edge detection, *IEEE Transactions on pattern analysis and machine intelligence*, no. 6, pp. 679–698, 1986.
- Dana, S., Addressing challenges in modeling of coupled flow and poromechanics in deep subsurface reservoirs, PhD thesis, The University of Texas at Austin, 2018.
- Dana, S., System of equations and staggered solution algorithm for immiscible two-phase flow coupled with linear poromechanics, *arXiv preprint arXiv:1912.04703*, 2019.
- Dana, S., Ganis, B., and Wheeler, M.F., A multiscale fixed stress split iterative scheme for coupled flow and poromechanics in deep subsurface reservoirs, *Journal of Computational Physics*, vol. **352**, pp. 1–22, 2018.
- Dana, S., Ita, J., and Wheeler, M.F., The correspondence between voigt and reuss bounds and the decoupling constraint in a two-grid staggered algorithm for consolidation in heterogeneous porous media, *Multiscale Modeling & Simulation*, vol. **18**, no. 1, pp. 221–239, 2020.
- Dana, S. and Jha, B., A fault slip model to study earthquakes due to pore pressure perturbations, *arXiv preprint arXiv:2104.06257*, 2021.
- Dana, S. and Wheeler, M.F., Convergence analysis of fixed stress split iterative scheme for anisotropic poroelasticity with tensor biot parameter, *Computational Geosciences*, vol. **22**, no. 5, pp. 1219–1230, 2018a.

- Dana, S. and Wheeler, M.F., Convergence analysis of two-grid fixed stress split iterative scheme for coupled flow and deformation in heterogeneous poroelastic media, *Computer Methods in Applied Mechanics and Engineering*, vol. **341**, pp. 788–806, 2018b.
- Dana, S. and Wheeler, M.F., Design of convergence criterion for fixed stress split iterative scheme for small strain anisotropic poroelastoplasticity coupled with single phase flow, *arXiv preprint arXiv:1912.06476*, 2019.
- Dana, S., Zhao, X., and Jha, B., Two-grid method on unstructured tetrahedra: Applying computational geometry to staggered solution of coupled flow and mechanics problems, *arXiv preprint arXiv:2102.04455*, 2021.
- Foulger, G.R., Wilson, M.P., Gluyas, J.G., Julian, B.R., and Davies, R.J., Global review of human-induced earthquakes, *Earth-Science Reviews*, vol. **178**, pp. 438–514, 2018.
- Krizhevsky, A., Sutskever, I., and Hinton, G.E., Imagenet classification with deep convolutional neural networks, *Advances in neural information processing systems*, vol. **25**, pp. 1097–1105, 2012.
- Maini, R. and Aggarwal, H., Study and comparison of various image edge detection techniques, *International journal of image processing (IJIP)*, vol. **3**, no. 1, pp. 1–11, 2009.
- McCann, M.T., Jin, K.H., and Unser, M., Convolutional neural networks for inverse problems in imaging: A review, *IEEE Signal Processing Magazine*, vol. **34**, no. 6, pp. 85–95, 2017.
- Valentine, A.P. and Trampert, J., Data space reduction, quality assessment and searching of seismograms: autoencoder networks for waveform data, *Geophysical Journal International*, vol. **189**, no. 2, pp. 1183–1202, 2012.
- Yoon, C.E., O'Reilly, O., Bergen, K.J., and Beroza, G.C., Earthquake detection through computationally efficient similarity search, *Science advances*, vol. **1**, no. 11, p. e1501057, 2015.