Parallel Implementation of Gabor Wavelet Processing in PyTorch

Team: Huiran Duan, Zelin Wu

Supervisor: Prof. Dr. Manuel Günther, Roger Bavibidila

April 27, 2023

1 Introduction

Image processing is the extraction of information from image data and has a long history in research. One particular traditional approach of image processing is the application of Gabor filters [Daugman, 1985], which have been used for the task of face recognition and other face-related tasks [Wiskott et al., 1997, Zhang et al., 2005, Günther, 2012, Günther et al., 2012, Schneider et al., 2011]. Gabor wavelets are filters that are applied to images, and they model the first levels of the human visual system. Even when training deep networks, it has been shown that the learned filters have large similarities with Gabor wavelets [Krizhevsky et al., 2012]. Early approaches of incorporating Gabor wavelets directly into deep learning systems, however, only model parts of the whole processing chain [Luan et al., 2018].

There exists and open-source package[1] that implements the whole chain of Gabor wavelet processing for images in C++ with Python bindings. This package has been implemented by Manuel Günther some years ago as part of Bob [Anjos et al., 2012, Anjos et al., 2017], but it has been deactivated in the last version due to a restructuring and removal of all C++ implementations.[2] The goal of this Master project is to rewrite this package using PyTorch, so that it can be reactivated.

The documentation[3] of the package shows some use cases. First, a family of (typically 40) Gabor wavelets is defined that includes Gabor wavelets in various scales and orientations. Second, a given image is filtered with these 40 Gabor wavelets to produce 40 complex-valued filter responses. Third, Gabor jets [Wiskott et al., 1997, Günther, 2012] are extracted at certain locations in the image. Finally, these Gabor jets can be compared with dedicated similarity functions to achieve various goals such as landmark detection [Wiskott et al., 1997], disparity estimation [Günther et al., 2012] or classification [Schneider et al., 2011, Günther, 2012]. As an example, Gabor jets extracted in a grid graph structure can be used for face recognition [Günther et al., 2012], an example implementation is also available.[4]

The problem with the existing implementation is that it is done in C++ and without parallelization. Furthermore, the available implementation does not allow the incorporation of the Gabor filtering into modern deep learning methods. Since PyTorch allows parallelization on the GPU and includes everything required to perform image filtering, the task is to implement the Gabor wavelet processing in PyTorch. This includes to define the (complex-valued) filters as PyTorch `Module`, the extraction of Gabor jets, and the parallel implementation of their similarity computations. Additionally, the existing documentation needs to be updated, and it is required that all tests defined in the current package will pass with the new implementation. Preliminary implementations of the `Module` are available and can be used, including Gabor wavelets in spatial and frequency domain, as well as first implementations of the Gabor wavelet transform.

2 Assignment

Before starting, we first need to set up the working environment. Since we will use and extend functionality in Bob [Anjos et al., 2012], documentation on the package development in Bob is available online[5] and should be followed.

The first task is to implement Gabor wavelet processing in PyTorch. This can be done in spatial domain by defining convolution filters similar to (or deriving from) 2D convolutions. Additionally, an implementation

[1] `https://gitlab.idiap.ch/bob/bob.ip.gabor`

[2] `https://gitlab.idiap.ch/bob/bob.ip.gabor/-/issues/6`

[3] `https://www.idiap.ch/software/bob/docs/bob/bob.ip.gabor/master/index.html`

[4] `https://www.idiap.ch/software/bob/docs/bob/bob.example.faceverify/master/index.html`

[5] `https://www.idiap.ch/software/bob/docs/bob/bob.extension/stable/index.html`

in frequency domain shall be provided that speeds up processing when filters are large. This frequency domain implementation requires defining Gabor wavelets in frequency domain, transforming images using fast Fourier transform (FFT), multiplying images and filters in frequency domain, and perform an inverse FFT to the results.

The second task is to transform the complex-valued responses of the Gabor filters into Euler representations, i.e., magnitude and phase, which should also be implemented as a PyTorch `Module` and can serve as an activation function on the previous results.

Third, Gabor jets should be defined by extracting the 40 responses at certain (or all) locations in the image. If time allows, grid graphs of Gabor jets should be implemented, including the I/O functionality, preferably using HDF5.

Fourth, different similarity functions should be implemented in PyTorch. Ideally, comparisons should be implemented in parallel, for example, for comparing a Gabor jet from a specific location to Gabor jets of all other locations in the image. If possible, such an implementation should be parallelized onto the GPU, but the `multiprocessing` module can also be utilized. Implementing disparity estimation is part of this task.

Finally, the available documentation of the package needs to be adapted such that it reflects the new implementation. Available test cases should not be touched unless they do not make sense in the new PyTorch environment. It should be made sure that the tests pass with the new implementation. Additionally, Jupyter notebooks can be created to showcase the usage of the package. Possibly, the package is re-integrated into the Bob ecosystem at the Idiap Research Institute.

3 Schedule

Assuming 30 hours of work per week and a total of 15 ECTS with an average of 30 hours per ECTS, we arrive at a total workload of 15 weeks full-time. These should be distributed as follows.

Week 1-3 Setting up the work environment, installing all required tools, building a joint software design and interface.

Week 4-5 Implementing the Gabor wavelet family as an extension of the PyTorch `Conv2d` layer. Implementing the activation function to turn complex-valued responses into Euler representations.

Week 6-7 Implementing the Gabor wavelet processing in frequency domain using `torch.fft` for the FFT.

⇒ Milestone 1: An image can be filtered with a family of Gabor wavelets, and the result of the spatial domain transform and the frequency domain transform produce similar results.

Week 8-9 Implementing Gabor jets, including I/O functionality. Possibly implement them as `Module` in PyTorch.

⇒ Milestone 2: Gabor jets can be extracted from images, saved and loaded from HDF5 files.

Week 10-12 Implementing similarity functions of Gabor jets and Gabor graphs in parallel.

⇒ Milestone 3: Gabor jet disparities and other similarity functions are implemented.

Week 13-15 Updating the documentation of the package and improving the test cases.

⇒ Milestone 4: The documentation is up-to-date and all test cases pass.

If time allows Implement the Gabor wavelet processing with learnable wavelets that can be updated by gradient descent. Design a small end-to-end network and learning example that shows a proof of concept.

Optionally Implement grid graphs of Gabor jets as image representation. Update the face verification example to work with the new package.

Optionally Integrate the new package into the Idiap ecosystem including the continuous integration system.

The project is designed for three students, given our current progress in the implementation of the Gabor wavelets. I recommend to first design the API interface, which makes it easier to combine other tasks and work on them in parallel. Afterward, the spatial and frequency-domain implementation of Gabor wavelets can be done in parallel by two students, while the third is working on Gabor jets and their similarities. Subsequently, the documentation and the test cases shall be updated, before going to the optional steps. It is recommended to update the test cases while implementing the functionality.

Writing the project report is part of the Master project. As a template, the LaTeX thesis template from my webpage[6] should be used. I would recommend to start writing early and keep note of what was done when, and by whom. At the end of the project, there will be a joint presentation of the results in my research group.

[6] `https://www.ifi.uzh.ch/en/aiml/theses.html`

4 References

[Anjos et al., 2012] Anjos, A., El-Shafey, L., Wallace, R., Günther, M., McCool, C., and Marcel, S. (2012). Bob: a free signal processing and machine learning toolbox for researchers. In *ACM International Conference on Multimedia (ACMMM)*, pages 1449–1452.

[Anjos et al., 2017] Anjos, A., Günther, M., de Freitas Pereira, T., Korshunov, P., Mohammadi, A., and Marcel, S. (2017). Continuously reproducing toolchains in pattern recognition and machine learning experiments. In *International Conference on Machine Learning (ICML).*

[Daugman, 1985] Daugman, J. G. (1985). Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters. *Journal of the Optical Society of America Part A.*

[Günther, 2012] Günther, M. (2012). *Statistical Gabor graph based techniques for the detection, recognition, classification, and visualization of human faces.* PhD thesis, Technical University of Ilmenau.

[Günther et al., 2012] Günther, M., Haufe, D., and Würtz, R. P. (2012). Face recognition with disparity corrected Gabor phase differences. In *International Conference on Artificial Neural Networks (ICANN)*, pages 411–418. Springer.

[Krizhevsky et al., 2012] Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems (NIPS).*

[Luan et al., 2018] Luan, S., Chen, C., Zhang, B., Han, J., and Liu, J. (2018). Gabor convolutional networks. *IEEE Transactions on Image Processing.*

[Schneider et al., 2011] Schneider, H. J., Kosilek, R. P., Günther, M., Roemmler, J., Stalla, G. K., Sievers, C., Reincke, M., Schopohl, J., and Würtz, R. P. (2011). A novel approach to the detection of acromegaly: accuracy of diagnosis by automatic face classification. *The Journal of Clinical Endocrinology & Metabolism*, 96(7):2074–2080.

[Wiskott et al., 1997] Wiskott, L., Fellous, J.-M., Krüger, N., and von der Malsburg, C. (1997). Face recognition by elastic bunch graph matching. *Transactions on Pattern Analysis and Machine Intelligence*, 19:775–779.

[Zhang et al., 2005] Zhang, W., Shan, S., Gao, W., Chen, X., and Zhang, H. (2005). Local Gabor binary pattern histogram sequence (LGBPHS): A novel non-statistical model for face representation and recognition. In *International Conference on Computer Vision (ICCV)*. IEEE.