CS 535 - Multicore Programming - Project Proposal
A Fast Implementation of Adaptive Median Filter Using Multithreading

Mustafa Ghanim, Cengiz Emre Dedeağaç

Spring 2022

Median Filter (MF) is widely used in image processing applications, such as noise removal and increasing the compression performance [1–4]. Local 2D MF is useful with noisy images that have acceptable Peak Signal-to-Noise Ratio (PSNR) values. However, for severely noisy images and when higher PSNR values are required, we need more advanced filtering techniques that consider higher number of spatial correlations between the neighboring pixels.

An Adaptive Median Filter (AMF) uses multiple computation stages before deciding to change the center pixel value (if it is detected as noisy), or not changing it (if it is detected as a regular image pixel). Hence, a cost/performance trade-off rises between a local MF with lower computational cost and an AMF that consumes more memory, power, and runtime. Such comparisons become more important in real-time applications due to the required tight timing targets and resources limitations. In applications where the timing is the primary concern, MF is often more viable than AMF because of its speed. We want to increase the speed of AMF so that we can get better results under more strict time limitations.

In this project, we will explore the algorithm of Adaptive Median Filter (AMF) using multithreading techniques in C/C++. For example, runtime optimization can be done with a multicore system by dividing each frame into sub-frames and output each AMF-processed sub-frame in parallel.

Furthermore, Simultaneous Multithreading (SMT) can be done for sub-frames with different runtime, depending on the necessity of adaptive filter windows.

We are planning to use GCC compiler on Linux and libraries such as: *pthread*, *stdlib*, *stdio*, *string*, *stdint*, *errno*, *math*, *jpeglib*, and *opencv* for our C/C++ implementation. Intel VTune will be chosen as our profiler. For performance benchmark, sequential versions of AMF and MF can be used, as well as reported elapsed time results in the literature (if any found). As for the image dataset, we can use the popular 512x512 standard test images, and/or high resolution test images.

The planned timeline is:

- Research of methodology, implementation, and literature: **weeks 6-8**.
- Implementation with different designs, parameters, and hardware resources (if we have enough time left): **weeks 9-12**.
- Report and presentation preparation: **weeks 13-15**.

References

[1] H. Hwang and R. Haddad, "Adaptive median filters: new algorithms and results," *IEEE Transactions on Image Processing*, vol. 4, no. 4, pp. 499–502, 1995.

[2] G. George, R. M. Oommen, S. Shelly, S. S. Philipose, and A. M. Varghese, "A survey on various median filtering techniques for removal of impulse noise from digital image," in *2018 conference on emerging devices and smart systems (ICEDSS)*, pp. 235–238, IEEE, 2018.

[3] H. Soni and D. Sankhe, "Image restoration using adaptive median filtering," *Image*, vol. 6, no. 10, 2019.

[4] N. Salih, A. Abid, and C. Eswaran, "Efficient retinal image compression based on modified huffman algorithm," *International Journal of Engineering Research and Technology*, vol. 12, no. 7, pp. 942–948, 2019.