Image Processing

Image processing was chosen as a topic, and the following papers are being reviewed or summarized:

1. S.A. Karkanis, G.D. Magoulas, D.K. Iakovidis, D.E. Maroulis, N. Theofanous, “Tumor recognition in endoscopic video images”, 26‘’ EUROMICRO conf. Netherlands, pp. 423429,2000.
2. Z. Hocenski et al, Ceramic Tiles Failure Detection Based on FPGA Image Processing, IEEE International Symposium on Industrial Electronics (ISlE 2009), Seoul Olympic Parktel, Seoul, Korea July 5-8, 2009
3. [Ye Li](http://ieeexplore.ieee.org/search/searchresult.jsp?searchWithin=p_Authors:.QT.Ye%20Li.QT.&searchWithin=p_Author_Ids:37964176300&newsearch=true" \o "" \t "_blank) ; [Qingming Yao](http://ieeexplore.ieee.org/search/searchresult.jsp?searchWithin=p_Authors:.QT.Qingming%20Yao.QT.&searchWithin=p_Author_Ids:37409959700&newsearch=true" \o "" \t "_blank) ; [Bin Tian](http://ieeexplore.ieee.org/search/searchresult.jsp?searchWithin=p_Authors:.QT.Bin%20Tian.QT.&searchWithin=p_Author_Ids:37968848300&newsearch=true) ; [Wencong Xu](http://ieeexplore.ieee.org/search/searchresult.jsp?searchWithin=p_Authors:.QT.Wencong%20Xu.QT.&searchWithin=p_Author_Ids:37962274600&newsearch=true" \o "" \t "_blank)  [Fast double-parallel image processing based on FPGA](http://ieeexplore.ieee.org/xpl/articleDetails.jsp?tp=&arnumber=5983754&queryText%3Dfast+double+parallel+image+processing+FPGA),  [Vehicular Electronics and Safety (ICVES), 2011 IEEE International Conference on](http://ieeexplore.ieee.org/xpl/mostRecentIssue.jsp?punumber=5975664)  
   DOI: [10.1109/ICVES.2011.5983754](http://dx.doi.org/10.1109/ICVES.2011.5983754), Page(s): 97 - 102

## Tumor Recognition in Endoscopic Video Images using Artificial Neural Network Architectures

This paper describes a method for training an artificial neural network (Multi-factor Feed-forward Neural Network, hereafter MFNN) to recognize the presence of a tumor by windowing the pixels to 32x32 and 64x64, preprocessing the pixels to produce 4 types of statistical weights that are further processed by the trained network, which will indicate whether a tumor is detected.

The tumor detection process in detail uses the following recipe:

1. The basic statistics p(i,j) within a gray scale image is the probability of going from gray level i to gray level j by moving a certain distance at a certain angle. The possible angles are 0,45, 90 and 135 degrees, and the pixel distance is fixed as one pixel, M=32, and the statistics is calculated over sliding windows of sizes M by M and 2M by 2M. This is described as texture analysis.
2. 4 statistical measures are defined, Energy/angular momentum squared, correlation, inverse difference moment and entropy. These 4 are selected from 14 measures proposed by Haralick, listed as reference [3], but probably reference [2] or [4], as reference [3] is not authored by Haralick. The authors claim that the 4 selected measures are the best disctiminants.
3. The i,j statistical functions are decomposed into 4 wavelet channels.
4. There is a final 48 component feature vectors used as input for the MFNN, the paper is quite unclear about how these are obtained, the authors’ names are Greek, and the explanation is Greek to me! It is clear that the 4 statistical measures, multiplied by 4 angles result in 16 statistical measures. I can guess that each of these 16 are decomposed into 4 wavelet bases, and 3 of those are selected, resulting in 48 feature vectors, however, this is a guess, and I think the reviewers were asleep, given they let the paper be published being this unclear, and with incorrectly labeled references.
5. A 3 layer neural network with a 20 neuron configuration was found to be optimal, end of section 3 says 10 through 50 neurons were compared, and 20 seemed best, however the tables of results later in the paper focuses on 25 and 30 neurons. I am assuming this discrepancy comes from the authors graduate student determining 20 neurons is best, but too late to generate additional figures and still meet the deadline.
6. There are comparisons between using the 16 feature vectors before the wavelet transform as input, and the 48 post wavelet transform feature vectors, the main effect of the wavelet transform is that the MFNN neural network gets trained faster, and the effect of using a sliding window size of 64x64 is that the false positives can drop from 5% to 2.5%. Neither window size nor wavelet transform seems to have much effect on the detection rate, which is in the low 90% range.
7. The interesting information that was skipped was the correlation between the missed detections as the image was scanned across a tumor. If the missed detections are somewhat random, i.e. if frame n missed the tumor, which might have a 10% chance, is it still a 10% chance that frame n+1 will miss the tumor, or would the misclassifications have a strong spatial correlation? Overall, I learned a lot, and I think the science was great, but the writing was sloppy.

# Ceramic Tiles Failure Detection Based on FPGA Image Processing

## Abstract

This paper describes method for automating quality control of ceramic tiles in a manufacturing line. The system relies on a line scan camera, which captures images that are continuously fed to an FPGA, which will assign a quality level to each tile, with which the tiles get marked. The intention is to decrease cost and increase accuracy relative to inspections by humans.

## Summary of Defects

The types of defects that need to be detected are color differences of the surface, edge and corner defects, cracks, abrasion marks, surface texture abnormalities etc.

## Defect Detection Hardware System description

The system scans one line of 1024 pixels continuously, as the tile moves on a conveyer belt. The method described in this paper works on plain tiles, not on textured ones. A FPGA is preferable to a PC for speed reasons. Line scan camera delivers an output signal with Low Voltage Differential Signal (LVDS), and because the system was based on a Spartan 3 FPGA without a LVDS converter, one was designed by the authors, and incorporated in the system. The image is represented as 8 bit gray scale, and it is stored in a 1024 x 8bit block ram (BRAM) within the FPGA. A function controller within the FPGA informs a printer controller (also within the FPGA) of the tile quality class. The printer controller instructs a tile printer to create a mark on the tile consistent with the quality class, hopefully on the back side of the tile! The FPGA system also contains a camera controller and an XGA controller for displaying on a monitor. The detected pixel intensities don’t necessarily go from 0 to 255, the optimal thresholds levels might depend on tile color, and it seems like setting threshold levels at max – 10% and min + 10 percent would work well.

## Defect Detection Algorithm Description

The system is described as a Finite State Machine (FSM) with 6 states corresponding to where in the flow a given tile is. There is an algorithm to put the tile into the “edge defect” state, and another to put it into the “surface defect” state. The “init” state sets white and black thresholds, which are later used in the algorithms for defect detection.

## Performance

The FSM can operate with a clock frequency of about 90 MHz using a Xilinx Spartan-3 XCS200 FPGA, which results in about 6 times the speed of a PC based implementation of the same algorithm, when the PC was running windows XP with a T7300 processor, which is a Core 2 duo with 2GHz clock frequency.

# Fast Double-Parallel image processing based on FPGA

## Abstract

Paper investigates edge detection and median filter algorithms performed using DSP, PC and FPGA. By implementing the maximum possible parallelism, dividing the image into many small parts, even an entry level FPGA like a CYCLONE II on a DE2 development board can be more than 100 times faster than the PC and the DSP. The application is enhancing low resolution images acquired by traffic cameras, edge detection will show the outline of a vehicle.

## Description of Median Filter and Edge Detection Methods

A median filter replaces 1 pixel by the median value of its neighborhood. The paper authors implemented a median of medians within a 3x3 neighborhood, which they claim saves time. (I personally have implemented the same algorithm, it is guaranteed if the input is 1 through 9, that median of medians will return either 4, 5 (true median) or 6]. Medians of medians also has a neat Wikipedia page [https://en.wikipedia.org/wiki/Median\_of\_medians ]. A true median (3x3) implementation can be achieved by first sorting rows, then sorting column, and lastly extract the median of the cross diagonal that does not contain the min and max value. This would result in at least 50% higher propagation delay, and consume about double the number of FPGA resources). Edge detection is evaluated using the Prewitt method, applied separately with 8 directional kernels, and use the strongest of the 8 signals.

## Implementation

System on chip is based on the NIOS-II soft core CPU, splitting the image into 2 parts that are independently processes, the median filter is applied first to remove pixel noise, followed by edge detection, then the processed image fractions are placed in on chip memory prior to being recombined. The system overview flow chart is illustrated in figure 2, and the algorithm overview flow chart is described in figure 3. The authors note, like I commented before, that the median of medians will return a value from the 3 center bins, which works perfectly for pixel noise.

## Results and Analysis

In order to avoid boundary issues, each half of the image contains 4 extra rows, so every pixel is processed as would be expected if the image was nor split. The Cyclone II FPGA runs at 50MHz, and is more than twice as fast as a quad core Intel processor running at 2.66GHz, and 13 times faster than a TI DSP chip running at 600MHz. In terms of performance per clock cycle, the FPGA beats both processors with more than a factor of 100.

## Conclusions

The results are pretty impressive even with an entry level FPGA, the performance would be expected to be even better with a larger and faster FPGA. My personal opinion is that