2: 1 UPSAMPLING-DOWNSAMPLING IMAGE RECONSTRUCTION SYSTEM

Mihai Zaharescu
Polytechnic University of Bucharest

Ion I. Bucur
Polytechnic University of Bucharest

2:1 UPSAMPLING-DOWNSAMPLING IMAGE RECONSTRUCTION SYSTEM

Engineer **Mihai Cristian TĂNASE *(mihaicristian.tanase@gmail.com)***
Engineer **Mihai ZAHARESCU *(mihai.zaharescu@cti.pub.ro)***
Associate Professor PhD Eng. **Ion BUCUR *(ion.bucur@cs.pub.ro)***

Department of Computer Science and Engineering, Faculty of Automatic Control and Computers Science
University "Politehnica" of Bucharest, Splaiul Independenţei 313, Bucharest, 060042, Romania

ABSTRACT

In this paper the subject that is considered is an efficient way to send images over an unreliable network link, so that with a small bandwidth, the receiver obtains data at the best quality in an acceptable time.

This paper resumes a study that was made, presented in detail, and gives an interesting new view of the process of sending images over the Internet.

KEYWORDS: Image processing, image compression, progressive coding, transfer protocol, image file format

INTRODUCTION

The study that was made in this paper is based on *upsampling–downsampling* images at the 2:1 ratio, considering most of the standard filters.

The study also resulted in a system that is constructed of two parts: the server (producer) side and the client (consumer) side. Between this two parts can by any kind of network connection: local, Internet, etc.

The result of the study is an implementation of the system that uses *BMP (Bitmap format)* and constructs a pyramid (see below) of different sizes of the original image.

SYSTEM IMPLEMENTATION

System overview

Each one of the system components has a package (a set of unique upsampling–downsampling filters) that is used to recursively construct a pyramid.

The pyramid building algorithm is straightforward, it is based on successive downsampling operations followed by upsampling and encoding of the residuals.

The Figure 1 represents an example of the resulting pyramid.

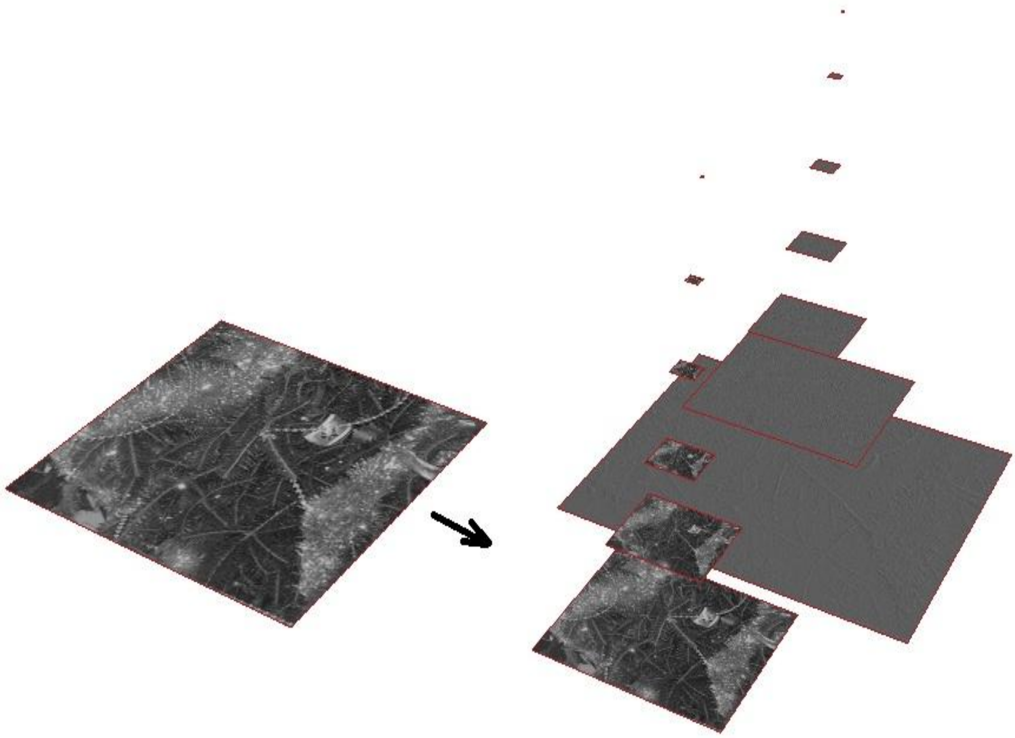

Figure 1. Constructing a pyramid of different image resolutions.

Each layer (level) of the pyramid is constructed by downsampling the layer below or upsampling the layer above and modifying the result with the residual image. The residual image is the difference between an image of a layer and the upsampled image of the layer above.

The system works as presented in the Figure 2.

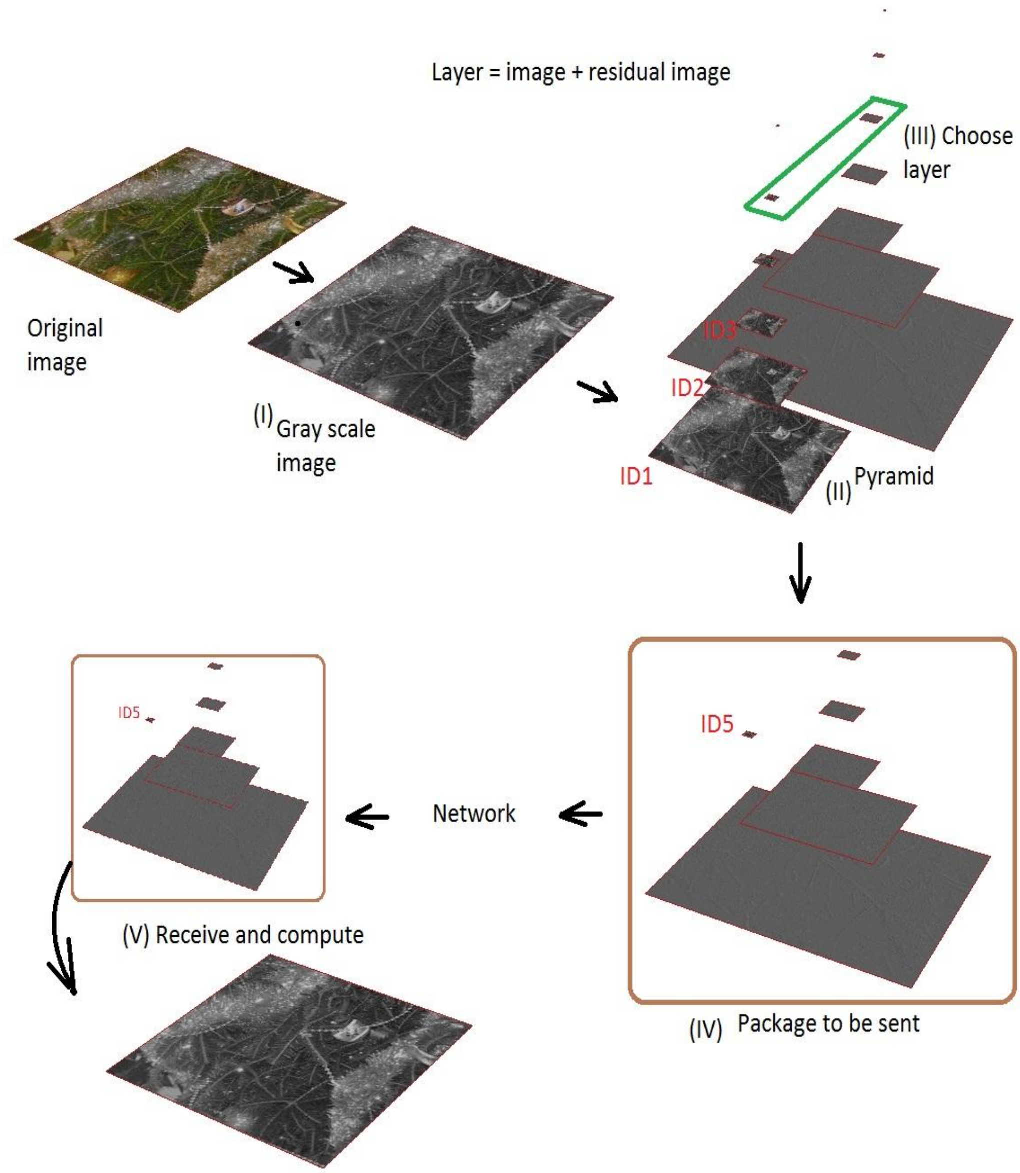

Figure 2. Phases in the image transformation system.

A layer of the pyramid contains the level number (starting from 1), the original image downsampled (resizing at half on both width and height) and the residual image, which is the difference between the original image and the upsampled image of this layer.

In this implementation, bandwidth variation is taken into consideration and it influences the chosen layer to be sent. The bandwidth is simulated by a random value between two chosen thresholds[1] (in the implementation, the lower threshold is 40 and the upper is

[1] The bandwidth is a value in kb (kilobytes).

5120). To select the layer, a linear interpolation is done between the upper, lower bandwidth threshold, current bandwidth value and 1 to highest number of the pyramids layers. The formula is:

$$S = (M - m) * (Ct - Lt) / (Mt - Lt)$$

where S is the selected layer Id; M the max layer Id; m the min layer Id; Ct the current threshold; Lt the lower threshold and Mt the max threshold.

In other words, when the bandwidth is close to the minimum accepted, the layer that is selected, when packaging (preparing) for sending, is close to the minimum.

The following step is represented by the sending process. In the implementation this is simulated by two different functions (one that fills up a package and the other that uses this package to do the next presented calculations). To fully generate the original image, a client (consumer) has to have all the residual images from the selected layer to the first, the bottom one.

The last step is the consumer side of the system, in which the package formed is used to generate recursively the originally-sized image.

Filters

In this system three filters are used: a box filter, a bilinear filter and one that generates pixels by multiplying. For downsampling a box filter is used by averaging four neighbors into a single pixel. For the upsampling the three interpolation methods mentioned are tested.

The residual image

On a given layer of the constructed pyramid, the residual image is the difference between the image of the lower level[2] and the upsampled image of the current layer. Because both images contain values from 0 to 255, the pixel values in the residual image are in the interval [-255, 255]. This is an important aspect as it will be used in the encoding process.

Of course, different filters (for upsampling or downsampling) will yield different residual images (see Figure 3).

[2] For the first layer of the pyramid, the lower layer is the original image.

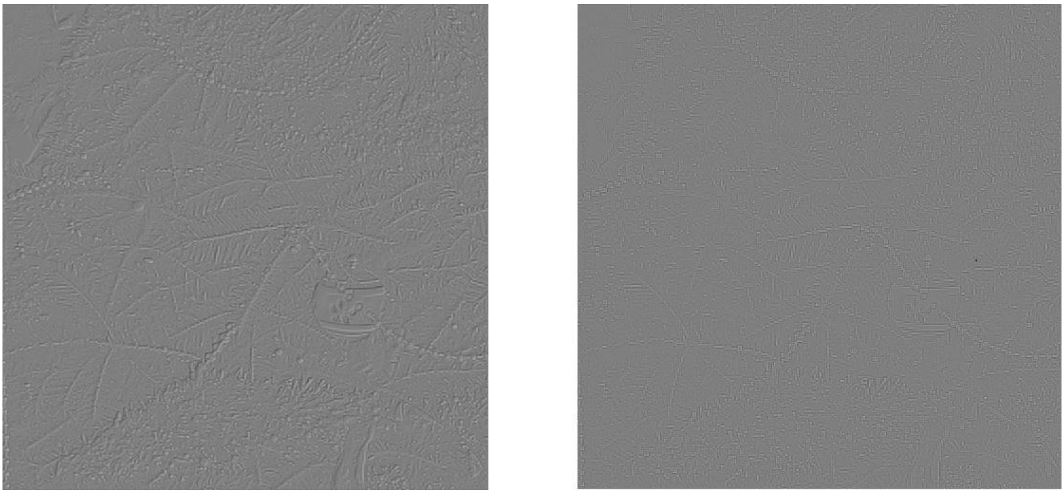

Figure 3. Comparison between two different upsampling filters: bilinear interpolation (left) and pixel multiplication (right)

One objective of the study is to select the best combination of filters that give the best results. The optimal result is directly influenced by the encoded size of the residual image.

In the implementation, for entropy coding purposes the Adaptive Huffman encoding was selected. [1]

TESTS AND RESULTS

When doing the tests, most of the parameters were altered (new images, different bandwidth, etc).

The graphs in Figure 4 show the original image size (the blue line) and the sizes of all the layers (values on the X-axis represent the number of downsamplings done, which are the same as the layer ID's in the pyramid).

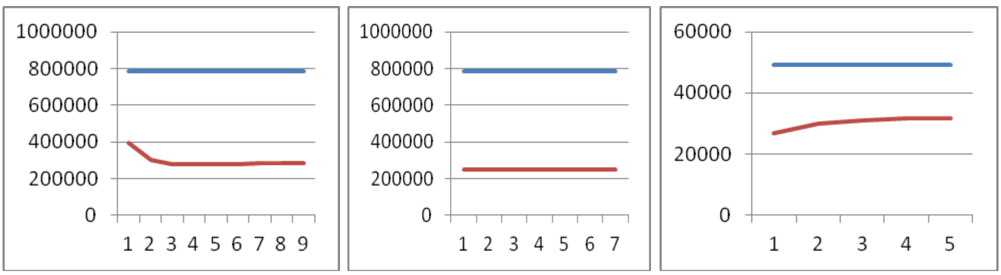

Figure 4. Comparison between sizes in the pyramids and original images
From left to right: ChristmasTree (512x512), ChristmasTree2 (512x512), Dinner (256x256)

In the Table 1 there are the results of the tests. One note here: going upward in the pyramid does not necessarily yield to a smaller package size.

Image	Bandwidth	Selected layer[3]	Size		
			Original image	Sent layer package	Size reduction (%)
ChristmasTree **(512x512)**	40	3	786486	288815	63.28
	2540	9	786486	311293	60.42
	5120	1	786486	399355	49.22
ChristmasTree2 **(512x512)**	40	4	786486	247118	68.58
	2540	2	786486	272330	65.37
	5120	1	786486	370151	52.94
Dinner **(256 x 256)**	40	2	49206	25890	47.38
	2540	1	49206	28630	41.82
	5120	7	49206	31688	35.60

Table 1. Test results

[3] The selected layer represents the number of down-samplings executed on the original image.

FUTURE WORK

In the study and, also, in the implementation, the priority was represented by the construction of a system capable of sending images at different resolutions in correspondence with the bandwidth. This is the reason for not having tested more resampling filters. So, the next step is to implement all the standard resampling filters and rerun the tests.

To obtain the most realistic results, the whole system has to be integrated and tested in a real world network.

The system can be adapted to video compression [3] and adaptive transmission. Correlation on frames in the already generated pyramid [2] can help introduce a warping transformation matrix that utilizes the redundancy between frames [9].

Instead of splitting the image one resolution at a time, we can split a resolution layer by the orientations of the features.[4][5] The advantage of this approach is that we can send a random orientation from each resolution layer, giving a first impression that the image has all the frequencies, and if there is time left we can send the remaining channels also.

Besides the lossless compression, a lossy compression, obtained by thresholding small values would offer good quality over a small band with network. A proposal over the traditional algorithms is using adaptive thresholding, adapted from binarization. [4][5] Whatever is black is split o n a different layer for sending it if there is spare time at the end. Clustering is another possibility for sending elements that generate visual impact first. [8]

All the filtering tested was done directly on the image, in spatial domain. But an idea is trying to filter the image in a different domain, like after applying the Radon transform [7] in order to minimize coherent edge deteriorations. The lines detected on the residual image can be sent in vectorized forms as a first a first iteration to give a fast impression of better resolution. [6]

CONCLUSIONS

This paper presents the implementation details and the results of a study which searches a more efficient way of sending images over various networks. It focuses on building a system that transforms images into their correspondent low resolution pairs, which leads to a structure similar to a pyramid.

Also, to further simulate the network environment, the system, takes an additional input, the bandwidth value. This implementation will need more refinement as stated in the "Further work" section, but, as shown in the "Tests and results" section it has an immediate benefit, by reducing, in some cases, close to 70%, the total size of the image.

ACKNOWLEDGEMENT

The authors would like to thank Costin-Anton Boiangiu for his original idea (*paper under review*) onto which the described system is based, for support and assistance with this paper.

REFERENCES

[1] Xudong Song, Yrjo Neuvo, "Image Compression Using Nonlinear Pyramid Vector Quantization", *Multidimensional Systems and Signal Processing*, Vol. 5, 1994, pp. 133-149.

[2] M. Sankar Kishore, K Veerabhadra Rao, "A study of correlation technique on pyramid processed images", *Sadhana*, Vol. 25, 2000, pp. 37-43.

[3] Luc Vandendorpe, Benoit Macq, "Optimum quality and progressive resolution of video signals", *Annales Des Télécommunications*, Vol. 45, 1990, pp. 487-502.

[4] Costin-Anton Boiangiu, Alexandra Olteanu, Alexandru Victor Stefanescu, Daniel Rosner, Alexandru Ionut Egner (2010). „Local Thresholding Image Binarization using Variable-Window Standard Deviation Response" (2010), *Proceedings of the 21st International DAAAM Symposium, 20-23 October 2010*, Zadar, Croatia, pp. 133-134.

[5] Costin-Anton Boiangiu, Andrei Iulian Dvornic. "Methods of Bitonal Image Conversion for Modern and Classic Documents". *WSEAS Transactions on Computers*, Issue 7, Volume 7, pp. 1081 – 1090, July 2008.

[6] Costin-Anton Boiangiu, Bogdan Raducanu. "Robust Line Detection Methods". *Proceedings of the 9th WSEAS International Conference on Automation and Information*, WSEAS Press, pp. 464 – 467, Bucharest, Romania, June 24-26, 2008.

[7] Costin-Anton Boiangiu, Bogdan Raducanu. "Effects of Data Filtering Techniques in Line Detection", *Proceedings of the 19th International DAAAM Symposium*, DAAAM International (Vienna, Austria), pp. 0125–0126, 2008.

[8] Costin-Anton Boiangiu, Bogdan Raducanu. "3D Mesh Simplification Techniques for Image-Page Clusters Detection". *WSEAS Transactions on Information Science, Applications*, Issue 7, Volume 5, pp. 1200 – 1209, July 2008.

[9] Costin-Anton Boiangiu, Daniel Rosner, Alexandra Olteanu, Alexandru Victor Stefanescu, Alin Dragos Bogdan Moldoveanu, "Confidence Measure for Skew Detection in Photographed Documents", *Annals of DAAAM for 2010, Proceedings of the 21st International DAAAM Symposium*, 20-23 October 2010, Zadar, Croatia, pp. 129-130.