IEEE International Workshop on Imaging
Systems and Techniques – IST 2007
Krakow, Poland, May 4–5, 2007

# A Log-Polar Interpolation Applied to Image Scaling

A. Amanatiadis[1], I. Andreadis[1], A. Gasteratos[2]

[1]Laboratory of Electronics, Department of Electrical and Computer Engineering,
Democritus University of Thrace, 12 Vas. Sofias Str, GR-67100, Xanthi, Greece
E–mail: {aamanat, iandread} @ee.duth.gr

[2]Laboratory of Robotics and Automation, Department of Production and Management Engineering,
Democritus University of Thrace, University Campus Kimmeria, GR-67100, Xanthi, Greece
E–mail: agaster@pme.duth.gr

***Abstract –*** *This paper proposes a bio-inspired interpolation algorithm suitable for image scaling. A log-polar neighbor model is adopted, utilizing the feature of applying larger weights to pixels at the center of the interpolation region and logarithmically decreasing weights to pixels away from the center. The interpolation is performed in the Cartesian plane without requiring the full transformation of the image to the log-polar plane. Experiments show that in both visual comparisons and quantitative analysis, the results extracted by the proposed log-polar neighbor model are better than those extracted from pixel repetition, bilinear and bicubic interpolation.*

***Keywords –*** *Log-polar interpolation, digital image scaling, log-polar neighbor model, zooming.*

## I. INTRODUCTION

Image interpolation is widely used in digital image processing tasks such as image scaling, image warping and image restoration. Various interpolation algorithms have been reported for image scaling. The simple scaling methods comprise zero order interpolation, as nearest neighbor interpolation and first order interpolation, as bilinear interpolation [1]. Both methods are widely used since they are cost-effective and realized in consumer electronics devices. However, the quality of the interpolated image is degraded by blocky artifacts, excessive smoothness or low performance in high frequency regions. Most refined methods such as spline interpolation have been also reported [2] but they require complex hardware implementations. Area based interpolation [3] computes each interpolated pixel by a proportional area coverage of a filtering window which is applied to the input image. The Mitchell and Lanczos [4] interpolation methods are not used widely and, additionally, are demanding in terms of hardware resources. Fractal zooming [5] has been recently reported with very good visual results but the computational cost for its coding remains its main problem.

Log-polar initial analytical formulation is based on studies on biological visual pathways from the retina to the visual cortex [6]. This bio-inspired model can be summarized to the following two assumptions. First, the distribution of the photoreceptors in the retina is not uniform. They lay more densely in the central region called fovea, while they are more sparse in the periphery. Consequently, the spatial resolution also decreases moving away from the fovea toward the periphery. It has a radial symmetry, which can be approximated by a polar distribution. Second, the projection of the photoreceptors array into the primary visual cortex can be well described by a logarithmic-polar (log-polar) distribution mapped onto a rectangular-like surface (the cortex). Various studies and applications based on log-polar image mapping have been reported. A vision application, estimating the time to impact, has been reported in [7], tracking using log-polar vision has been presented in [8], wavelet extraction based on log-polar mapping has been introduced in [9] and disparity estimation and vergence control of an active stereo vision system has been developed in [10].

In this paper, we propose a novel image interpolation algorithm for image scaling based on log-polar neighbor mapping. The log-polar neighbor model is adopted, exploiting its previously mentioned attributes, by applying larger weights to pixels at the center of the interpolation region and logarithmically decreasing weights to the peripheral pixels. The aim of the proposed interpolation is to preserve the geometric regularity around edges and generate interpolated images with higher visual quality, without the need of additional processing for edge identification. The algorithm can be applied to both grayscale and color images for any scaling factor. For color images, the algorithm is simply applied to each component separately. Both visual and quantitative simulation results have shown a fine high-frequency response and low interpolation error compared to other widely used algorithms.

The rest of the paper is organized as follows. In Section II, we describe in details the proposed interpolation algorithm. In Section III, we present the algorithm's internal parameters and their influence to algorithm's performance. In Section IV, we report the simulation results and compare the performance of the proposed algorithm with other widely used interpolation algorithms through quantitative measurements. We give concluding remarks in section V.

1-4244-0965-9/07/$25.00 ©2007 IEEE

## II. PROPOSED ALGORITHM

Let $S(i,j)$ be the input image with $m,n$ dimensions and $s$ the scale factor. For the scaling up process, we first define the output image $T(k,l)$, where $k = 1,2,\ldots,s \times m$ and $l = 1,2,\ldots,s \times n$. In case of $s \times m$ and $s \times n$ are not integers, they are rounded to the nearest integer. The intensity values of all pixels in $T$ are left undefined. Next, we map each output pixel onto the input image. The $x$ and $y$, Cartesian coordinates of the center of the mapped output pixel, onto the input image are calculated using the following equations

$$x_0 = \frac{l+0.5}{s},\ y_0 = \frac{k+0.5}{s} \tag{1}$$

where $k = 1,2,\ldots,s \times m$ and $l = 1,2,\ldots,s \times n$, assuming that each pixel has a height and width equal to one. The applied circular mask, on the input image, with $x_0, y_0$ center coordinates and radius $\rho_{max}$ is the interpolation region. The pixels, within the interpolation region, will contribute to the final calculation of the output pixel intensity, as shown in Fig.1. Next, we map the pixels in the interpolation region using the log-polar distribution. Since the pixels in Cartesian coordinates cannot be mapped one-to-one onto the pixels in the log-polar coordinate space, we apply an inner resampling of nearest neighbor. The relationship between the polar coordinates $(\rho,\theta)$ used to map the pixels of the interpolation region and the Cartesian plane $(\xi,\eta)$, can be described by [11]

$$\xi = log_a \frac{\rho}{\rho_0} \quad \text{if} \quad \rho_{max} > \rho > \rho_0 \tag{2}$$

$$\eta = q \times \theta \tag{3}$$

where $\rho_{max}$ is the maximum radius of the interpolation region, $\rho_0$ is the radius of the innermost circle, $1/q$ is the minimum angular resolution of the log-polar layout and $(\rho,\theta)$ are the polar coordinates. Subsequently, the coordinates of the mapped input image pixels $S(x_i, y_j)$ in the interpolation region are computed using the following equations

$$x_i = [\rho \times cos(\theta) + x_0] \tag{4}$$

$$y_j = [\rho \times sin(\theta) + y_0] \tag{5}$$

square brackets indicate the integer part expressed by the inner resampling of the nearest neighbor. The different weight is expressed by the number of times an input pixel will be mapped onto the interpolation region by the log-polar distribution. Input pixels inside the inner cycles will have more weight, as they are mapped more times than the pixels in the outer circles. Finally, the intensity of the output pixel is given by

$$T(k,l) = \frac{1}{c}\sum_{1}^{c} S(x_i, y_j) \tag{6}$$

where $c$ expresses the aggregate sum of the number of times all input pixels have been mapped within the interpolation region. The scaling down process is described by exactly the same procedure with scale factor varying from 0 to 1.

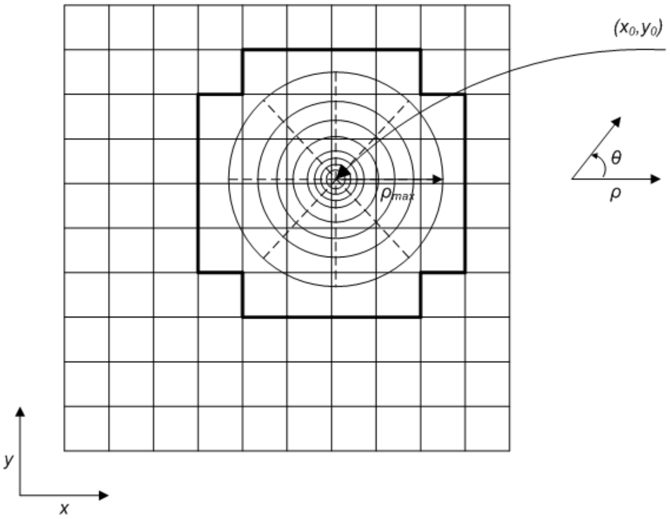


Fig. 1. Part of the input image $S$ and the log-polar interpolation neighborhood. The input pixels within the bold black line will contribute to the final output pixel intensity $T(k,l)$.

## III. INTERNAL PARAMETER SELECTION

Despite the fact that the algorithm is simple and straightforward, several internal parameters affect its accuracy and performance. The selection of the optimal numerical values of the parameters were chosen by extensive testing procedures and trade-offs. The parameters that were sustained to this evaluation are the maximum length of the radius of the interpolation region $\rho_{max}$ in (2), the radius resolution of the interpolation region, the minimum angular resolution of the log-polar layout $1/q$ in (3) and the logarithm's base $a$ in (2). The accuracy of the interpolation algorithm depends on how accurately these parameters are defined.

Depending on the length of the radius of the interpolation region $\rho_{max}$, the number of maximum contributing input pixels varies. Due to the log-polar interpolation all contributing input pixels are not of the same weight. More precisely the contributing input pixels of the innermost circles will contribute more than the ones of the external circles, inside the interpolation area. It is apparent that as the radius increases, more input pixels contribute to the final output pixel. Small radius means a small number of contributing input pixels, which formulate an algorithm with low computational burden, but also with low accuracy and inefficient spatial information. However, a large interpolation radius with a large amount of contributing input pixels, forms a more computationally demanding algorithm with redundant spatial information which involve blurring. In order to study how the change in interpolation radius is affecting the algorithm's efficiency, we used the Root Mean Square Error (RMSE). The tests, that were carried out in several images, utilize the reconstruction RMSE of an initially scaled up image and then scaled down by the same scaling factor. The RMSE between the reference image $I(x,y)$

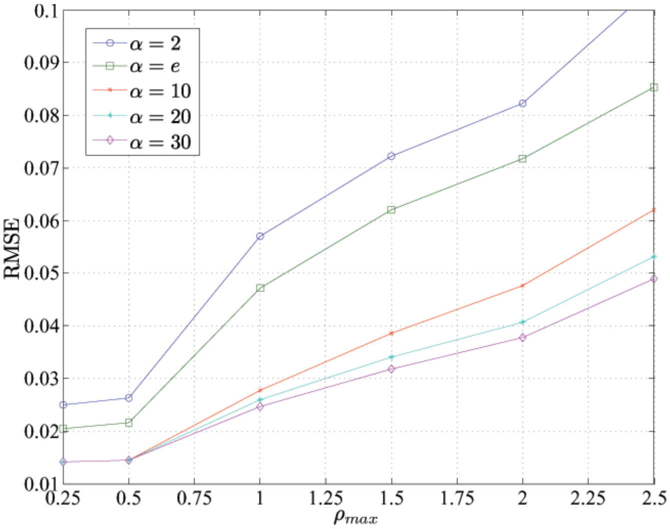


Fig. 2. Average RMS errors with respect to different radius $\rho_{max}$ of the interpolation region in different logarithm bases $\alpha$.

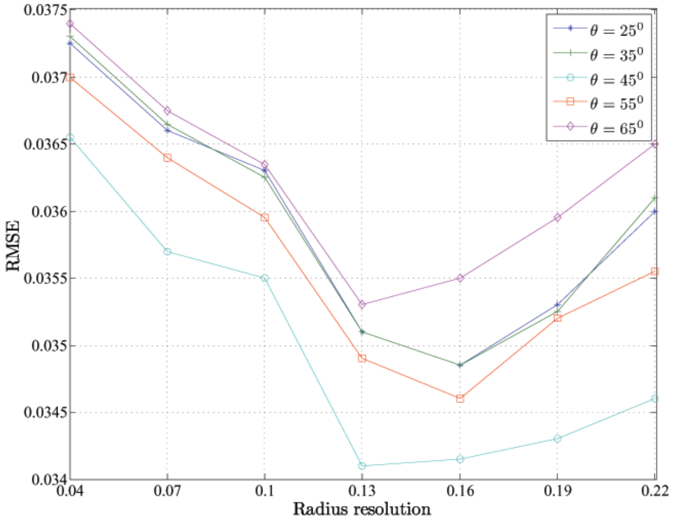


Fig. 3. RMS errors versus radius resolution in different angular resolutions.

and the reconstructed interpolated image $\hat{I}(x, y)$ is given by

$$e_{rms} = \left( \frac{1}{m \times n} \sum_{x=0}^{m-1} \sum_{y=0}^{n-1} \left( \hat{I}(x, y) - I(x, y) \right)^2 \right)^{1/2} \tag{7}$$

where the images are of size $m \times n$.

In Fig.2, we can see the RMSE for reconstructing an input image applying different $\rho_{max}$ for several logarithm bases $\alpha$, with same angular $1/q$ and radius resolution. Several test images were used and the average RMSE is shown for each different radius. As expected, the error goes up with larger radius, as more input pixels are contributing. The increase is gradual since the number of times extra input pixels are mapped by (4) and (5) is also gradually increased. However, the choice of a radius lower than $0.25$, where the lowest RMSE is observed, is not the ideal while only one input pixel is mostly contributing, arising the blocky effect which is observed also in the nearest neighbor interpolation. Radius ranging from $0.5$ and more is not the best solution since the error and computational burden are increasing to undesirable levels. Furthermore, the arisen blurring deteriorates the spatial quality of the output image.

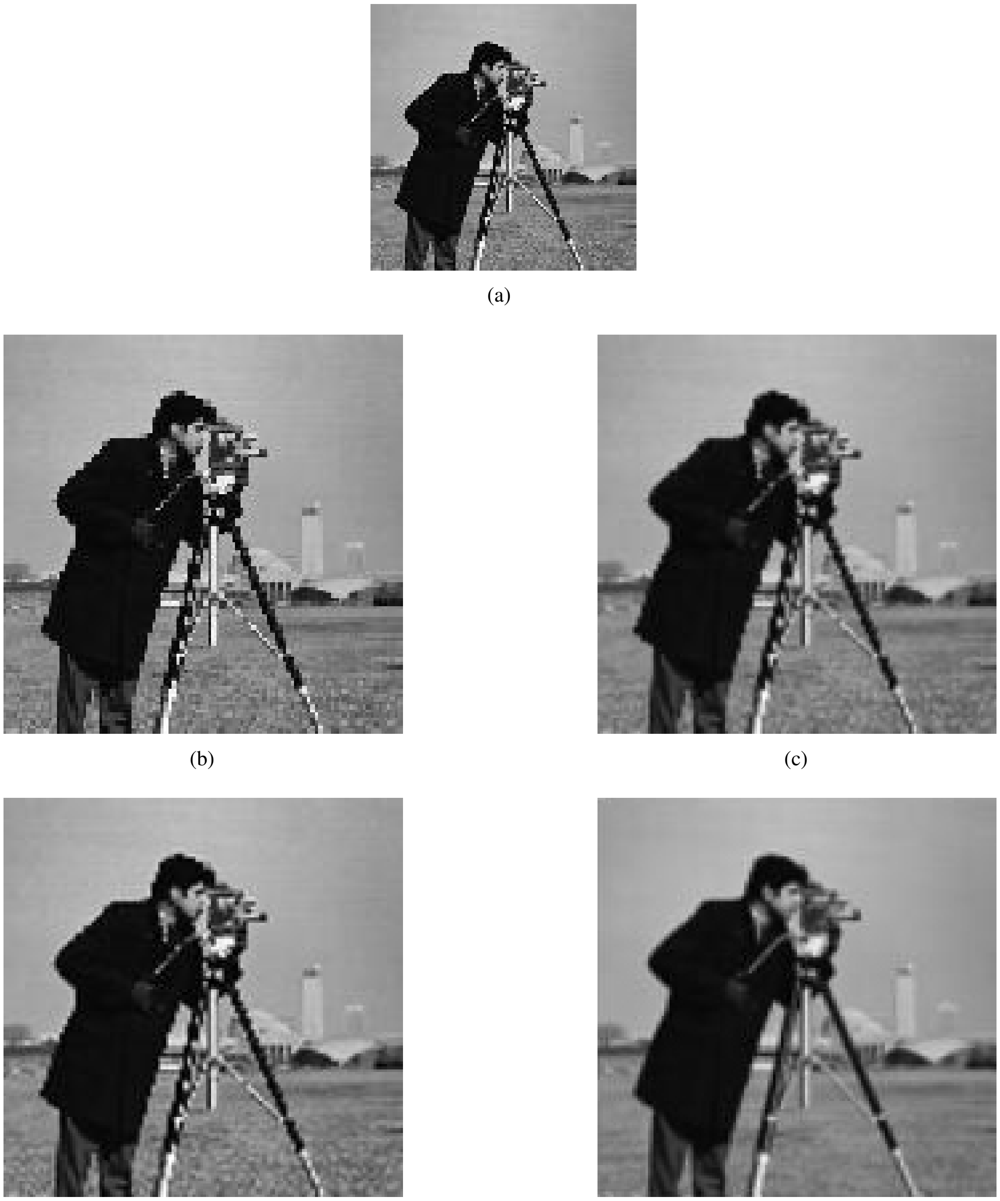
The proposed bio-inspired model is mainly described by the logarithm function in (2). As shown in Fig.2 the performance significantly changes with different base in the logarithm function. The algorithm achieves the lowest RMSE levels for a logarithm base of more than 10. Increasing the base affects the function output in a way that is desirable by the interpolation scheme. More precisely, the increment in the base improves the feature of applying even larger weights to pixels of the center of the interpolation region as well as decreasing even more the weights to pixels away from the center.

In order to quantify the performance of the proposed method in different angular and radius resolutions we compared the reconstruction RMSE by scaling the same image, with constant $\rho_{max}$ and logarithm base $\alpha$, in different angular and radius resolutions. Fig. 3 shows the calculated errors versus radius resolutions in different angular resolutions. We can notice that angular resolution of $45^0$ introduces the best performance. The general behavior that we described before is confirmed also by this plot. Small angular resolutions increase the overall input pixel contribution in the same interpolation region. The contribution is increased not only to input pixels of the inner circles but also to pixels of the external circles. However, when the angular resolution is very large, many input pixels are not contributing at all, while they are not mapped by (4) and (5). This suggests that an angular resolution of $45^0$ is the suitable value as an input argument.

By inspection of the RMSE of the proposed method in different radius resolutions, taking into account that each pixel has a height and width equal to one, it can be seen that the variation in the error is not affected significantly. However, while the resolution is low, varying from $0.04$ to $0.13$, the error is high but mostly the computational time is also increased drastically. For resolutions more than $0.16$, insufficient spatial information is used, as the inner circle input pixels of the interpolation area are not mapped at a desirable amount of times. It is interesting to notice that further increase to radius resolution results to additional increment of the error as shown in Fig.3. Concluding, based on all the above results, we observed that a maximum radius of $0.25$ to $0.5$, an angular resolution of $45^0$, a radius resolution of $0.13$ to $0.16$ and a logarithm base of more than $10$, produce the best results without making the algorithm computationally complex and time demanding.

## IV. EXPERIMENTAL RESULTS

In order to evaluate the performance of the proposed algorithm we performed several comparison tests. Initially, we



(a)

(b)

(c)

(d)

(e)

Fig. 4. Interpolation results for cameraman: (a) input image; (b) nearest interpolation; (c) bilinear interpolation; (d) bicubic interpolation; (e) proposed interpolation.

demonstrate the visual results, scaling up the "cameraman" image by a scaling factor of 1.5. The results are shown in Fig. 4 compared to the standard algorithms of the nearest neighbor, bilinear interpolation and bicubic interpolation. The maximum interpolation area radius used is $\rho_{max} = 0.5$, the angular resolution $\theta = 45^0$, radius resolution $0.15$ and logarithm base $a = 10$. We can notice that more jagged contours are dramatically produced in the scaled image by nearest neighbor interpolation in Fig.4(b). The Fig.4(e) has better visual quality, since the proposed interpolation better preserves the geometric regularity around edges and thus generates interpolated images with higher visual quality.

Although the visual results show a good performance, it is well established that visual results can not provide decisive indication of the quality of the scaled images. Thus, we evaluated the interpolation RMSE, in several images, by a testing procedure of initially scaling down and then scaling up the image, by a scaling factor of 2, in order to obtain the original size. Results, in Table I, demonstrate an overall superiority of the proposed method against the interpolations of nearest neigh-

TABLE I

RMSE'S PERFORMANCE COMPARISON

| **Test image** | **Interpolation method** | | | |
|---|---|---|---|---|
| | **Nearest** | **Bilinear** | **Bicubic** | **Proposed** |
| Lena | 0.0448 | 0.0293 | 0.0236 | 0.0178 |
| Cameraman | 0.0761 | 0.0556 | 0.0484 | 0.0385 |
| Tire | 0.0542 | 0.0337 | 0.0263 | 0.0209 |
| Peppers | 0.0296 | 0.0188 | 0.0153 | 0.0113 |

bor, bilinear and bicubic, in all test images.

## V. CONCLUSION

A new bio-inspired interpolation method for scaling has been introduced. It features a new log-polar neighbor model utilizing its attribute for applying larger weights to pixels of the center of the interpolation region and logarithmically decreasing weights to pixels away from the center. An accurate selection of the internal parameters has been presented for achieving the best accuracy and performance. The algorithm can be applied to both grayscale and color images for any scaling factor. Visual and quantitative experimental results have shown a fine high-frequency response and low interpolation error compared to other widely used algorithms.

## REFERENCES

[1] R. Gonzalez and R. Woods, *Digital Image Processing*. Addison-Wesley Reading, Mass, 1987.

[2] J. Hwang and H. Lee, “Adaptive image interpolation based on local gradient features,” *IEEE Signal Processing Letters*, vol. 11, no. 3, pp. 359–362, 2004.

[3] I. Andreadis and A. Amanatiadis, “Digital image scaling,” *Proceedings of IEEE Instrumentation and Measurement Technology Conference*, vol. 3, pp. 2028–2032, 2005.

[4] G. Grevera and J. Udupa, “Shape-based interpolation of multidimensional grey-level images,” *IEEE Trans. on Medical Imaging*, vol. 15, no. 6, pp. 881–892, 1996.

[5] D. Giusto, M. Murroni, and G. Soro, “Slow motion replay of video sequences using fractal zooming,” *IEEE Trans. on Consumer Electronics*, vol. 51, no. 1, pp. 103–111, 2005.

[6] U. Schwarz, C. Busettini, and F. Miles, “Ocular responses to linear motion are inversely proportional to viewing distance,” *Science*, vol. 245, no. 4924, pp. 1394–1396, 1989.

[7] M. Tistarelli and G. Sandini, “On the advantages of polar and log-polar mapping for direct estimation of time-to-impact from optical flow,” *IEEE Trans. on Pattern Analysis and Machine Intelligence,*, vol. 15, no. 4, pp. 401–410, 1993.

[8] G. Metta, A. Gasteratos, and G. Sandini, “Learning to track colored objects with log-polar vision,” *Mechatronics*, vol. 14, no. 9, pp. 989–1006, 2004.

[9] C. Pun and M. Lee, “Log-polar wavelet energy signatures for rotation and scale invariant texture classification,” *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 25, no. 5, pp. 590–603, 2003.

[10] R. Manzotti, A. Gasteratos, G. Metta, and G. Sandini, “Disparity estimation on log-polar images and vergence control,” *Computer Vision and Image Understanding*, vol. 83, no. 2, pp. 97–117, 2001.

[11] F. Jurie, “A new log-polar mapping for space variant imaging. Application to face detection and tracking,” *Pattern Recognition*, vol. 32, no. 5, pp. 865–875, 1999.