HBRP PUBLICATION

Journal of Emerging Trends in Electrical Engineering
Volume 4 Issue 1
DOI: [To be assigned]

Determinant based Fully Automatic one Scan Adaptive Image Scaling Algorithm

Sivarajan R.[1]*, Selvam K.[2], Nagarajan V.[3], Vedasundaravinayagam P.[4]
[1,4]Assistant Professor, Adhiparasakthi Engineering College, Melmaruvathur, Tamil Nadu, India
[2]P.G. Scholar, Adhiparasakthi Engineering College, Melmaruvathur, Tamil Nadu, India
[3]Professor, Adhiparasakthi Engineering College, Melmaruvathur, Tamil Nadu, India
****Corresponding Author***
E-Mail Id: sivarajanbeme@gmail.com

ABSTRACT
Image Scaling is defined as the resizing of images either by means of up-scaling or down-scaling. Various Algorithm such as Nearest-neighbor interpolation, Bilinear and bicubic algorithms, Sinc and Lanczos resampling, Box sampling, Mipmap, Fourier-transform methods, Edge-directed interpolation, hqx, Vectorization and Deep convolutional neural networks are available to perform image scaling. Carlo Arcelli et al. proposed a fully automatic one scan adaptive image scaling algorithm in which expanded pixel is found by the sum of weighted neighboring pixel. This paper proposes a same fully automatic one scan adaptive image scaling algorithm in which expanded pixel is found by the determinant of weighted neighboring pixel instead of summation. Finally both the existing and proposed algorithm is compared by means of PSNR and SSIM.

Keywords: *Algorithm, PSNR, SSIM*

INTRODUCTION

Image zooming algorithm plays a vital role in various application especially in medical field. For example if the doctor wants to check whether the patient is affected by ulcer then the doctor will take scan on the stomach in which affected area will be clearly viewed by the means of zooming process.

If the image is broken when the scanned image is zoomed then the doctor will not correctly predict whether the patient is affected by ulcer or not. Hence it is required to get better image quality when the image is zoomed. The better image quality can be recognized by means of PSNR and SSIM value.

The various basic algorithms used for performing image scaling are Nearest-neighbor interpolation, Bilinear and bicubic algorithms, Sinc and Lanczos resampling, Box sampling, Mipmap, Fourier-transform methods, Edge-directed interpolation, hqx, Vectorization and Deep convolutional neural networks. In nearest neighbor interpolation, each pixel in the zoomed version is replaced by its neighboring pixel and we can preserve sharp details nut introduces the jaggedness in the zoomed image.

In bilinear and bicubic algorithm, it introduces continuous transition for interpolating pixel color values into the output (zoomed image) even the original image has discrete transition. In Sinc resampling, there is a better reconstruction for zoomed image for a perfectly band-limited signal.

An approximation of the Sinc method called Lanczos resampling which produces better result compared to Sinc resampling. Also there is a computationally efficient

HBRP PUBLICATION

Journal of Emerging Trends in Electrical Engineering
Volume 4 Issue 1
DOI: [To be assigned]

approximation of the Lanczos resampling method is called Bicubic interpolation.

The methods till now discussed to perform image zooming has a limitation that to sample only the limited number of pixel. To avoid this, we introduce box sampling in which the algorithm samples non adjacent pixel when the down scaling below a certain threshold value. The threshold value is the twice for all bi-sampling algorithms. The optimization technique is very hard to done for this algorithm.

To solve this problem, another method called mipmap which is a prescaled set of downscale copies. This algorithm ensures that there is no scaling happens below the threshold of bilinear scaling. To preserve the edges of the zoomed image, edge directed interpolation is used. Also, image scaling can be done using vectorization and deep convolutional neural network.

Carlo Arcelli et al proposes the fully automatic one scan adaptive image scaling algorithm. This paper shows the WAZ algorithm is best compared to other interpolation methods such as nearest neighbor, bilinear and bicubic interpolation, etc. This WAZ algorithm is more convenient for performing image scaling compared to ALZ.[1-3]

EXISTING ALGORITHM – WEIGHTED SUM

The algorithm for the fully automatic one scan adaptive image scaling algorithm is shown below:

Step 1: Load the input color image.
Step 2: Separate the three color components R, G, and B plane from the color image.
Step 3: Identify the eight neighbors of each pixel in each color components.
Step 4: Expand each pixel for n x n matrix named it as ‘q’ using weighted sum.

For the zooming factor of 2,

$$q(1,1) = 4*p + 1*tl + 2*t + 2*l$$
$$q(1,2) = 4*p + 2*t + 1*tr + 2*r$$
$$q(2,1) = 4*p + 2*l + 1*bl + 2*b$$
$$q(2,2) = 4*p + 2*b + 1*br + 2*r$$

For the zooming factor of 3,

$$q(1,1) = 9*p + 6*l + 6*t + 4*tl$$
$$q(1,2) = 9*p + 3*l + 6*t + 2*tl + 2*tr + 3*r$$
$$q(1,3) = 9*p + 6*t + 4*tr + 6*r$$
$$q(2,1) = 9*p + 6*l + 3*t + 2*tl + 3*b + 2*bl$$
$$q(2,2) = 9*p + 3*l + 3*r + 3*b + bl + br + 3*t + tl + tr$$
$$q(2,3) = 9*p + 6*r + 3*b + 2*br + 3*t + 2*tr$$
$$q(3,1) = 9*p + 6*l + 4*bl + 6*b$$
$$q(3,2) = 9*p + 3*l + 3*r + 2*bl + 6*b + 2*br$$
$$q(3,3) = 9*p + 6*r + 6*b + 4*br$$

For the zooming factor of 4,

$$q(1,1) = 6*l + 4*tl + 6*t + 9*p$$
$$q(1,2) = 3*l + 2*tl + 8*t + 12*p$$
$$q(1,3) = 8*t + 2*tr + 3*r + 12*p$$
$$q(1,4) = 6*t + 4*tr + 6*r + 12*p$$
$$q(2,1) = 8*l + 2*tl + 3*t + 12*p$$

$$q(2,2) = 4 * l + tl + 4 * t + 16 * p$$
$$q(2,3) = 4 * t + tr + 4 * r + 16 * p$$
$$q(2,4) = 12 * p + 8 * r + 2 * tr + 3 * t$$
$$q(3,1) = 12 * p + 3 * b + 2 * bl + 8 * l$$
$$q(3,2) = 16 * p + 4 * b + bl + 4 * l$$
$$q(3,3) = 16 * p + 4 * b + br + 4 * r$$
$$q(3,4) = 12 * p + 3 * b + 2 * br + 8 * r$$
$$q(4,1) = 9 * p + 6 * l + 4 * bl + 6 * b$$
$$q(4,2) = 12 * p + 3 * l + 2 * bl + 8 * b$$
$$q(4,3) = 12 * p + 8 * b + 2 * br + 3 * r$$
$$q(4,4) = 9 * p + 6 * b + 4 * br + 6 * r$$

where p is the value of the pixel, and l, r, t, b, tl, tb, bl, and br are the neighbors of the pixel p.

Step 5: After computing q matrix find the maximum and minimum value in the q matrix.
Step 6: Find the value of Δ by using the formula

$$\Delta = \min_{q_k \neq p}\{|q_k - p|\}$$

Step 7: Find the value of T by using the formula

$$T = \begin{cases} q_{max} + q_{min} - p & if\ (q_{max} + q_{min})/2 \neq p \\ p & if\ (q_{max} + q_{min})/2 = p \end{cases}$$

Step 8: Find the final value of the pixel in q matrix is obtained by

$$q_k = \begin{cases} q_k - \Delta & if\ p \leq q_{min}\ for\ any\ q_k > p \\ q_k + p & if\ p \geq q_{max}\ for\ any\ q_k < p \\ \min(255, q_k + \Delta) & if\ q_{min} < p < q_{max}\ for\ any\ q_k \geq T \\ \min(0, q_k - \Delta) & if\ q_{min} < p < q_{max}\ for\ any\ q_k < T \end{cases}$$

Step 9: Finally calculate the PSNR and SSIM between the input and zoomed algorithm.

$$PSNR = 20\log_{10}\left(\frac{255}{\sqrt{MSE}}\right)$$

where MSE is the mean square error.

$$SSIM = \frac{(2\mu_v\mu_w + c_1)(2cov_{vw} + c_2)}{(\mu_v^2 + \mu_w^2 + c_1)(\sigma_v^2 + \sigma_w^2 + c_2)}$$

μ_v is the mean of the input image
μ_w is the mean of the zoomed image
σ_v^2 is the variance of the input image
σ_w^2 is the variance of the zoomed image
$c_1 = (k_1L)^2$ and $c_2 = (k_2L)^2$
$k_1 = 0.01$; $k_2 = 0.03$ and $L = 255$

The Pseudo code for the fully automatic one scan adaptive image scaling algorithm is shown below:

```
P ← Input image
Q ← Output image
N ← zooming factor
for each pixel in p do
        tl=p(i-1,j-1,k);
t=p(i-1,j,k);
```

```
tr=p(i-1,j+1,k);
l=p(i,j-1,k);
p=p(i,j,k);
r=p(i,j+1,k);
bl=p(i+1,j-1,k);
b=p(i+1,j,k);
br=p(i+1,j+1,k);
end for
for each pixel in p do
for i = 1 to n do
if N=2 do
q(1,1)=4*p+1*tl+2*t+2*l;
q(1,2)=4*p+2*t+1*tr+2*r;
q(2,1)=4*p+2*l+1*bl+2*b;
q(2,2)=4*p+2*b+1*br+2*r;
elseif N=3 do
q(1,1)=9*p+6*l+6*t+4*tl;
q(1,2)=9*p+3*l+6*t+2*tl+2*tr+3*r;
q(1,3)=9*p+6*t+4*tr+6*r;
q(2,1)=9*p+6*l+3*t+2*tl+3*b+2*bl;
q(2,2)=9*p+3*l+3*r+3*b+bl+br+3*t+tl+tr;
q(2,3)=9*p+6*r+3*b+2*br+3*t+2*tr;
q(3,1)=9*p+6*l+4*bl+6*b;
q(3,2)=9*p+3*l+3*r+2*bl+6*b+2*br;
q(3,3)=9*p+6*r+6*b+4*br;
else // N=4
q(1,1)=6*l+4*tl+6*t+9*p;
q(1,2)=3*l+2*tl+8*t+12*p;
q(1,3)=8*t+2*tr+3*r+12*p;
q(1,4)=6*t+4*tr+6*r+12*p;
q(2,1)=8*l+2*tl+3*t+12*p;
q(2,2)=4*l+tl+4*t+16*p;
q(2,3)=4*t+tr+4*r+16*p;
q(2,4)=12*p+8*r+2*tr+3*t;
q(3,1)=12*p+3*b+2*bl+8*l;
q(3,2)=16*p+4*b+bl+4*l;
q(3,3)=16*p+4*b+br+4*r;
q(3,4)=12*p+3*b+2*br+8*r;
q(4,1)=9*p+6*l+4*bl+6*b;
q(4,2)=12*p+3*l+2*bl+8*b;
q(4,3)=12*p+8*b+2*br+3*r;
q(4,4)=9*p+6*b+4*br+6*r;
end if
end for
if (qk ≠ p) do
Δ = min (|qk – p|)
end if
if (q max + qmin)/2 ≠ p do
t = q max + qmin – p
```

Journal of Emerging Trends in Electrical Engineering
Volume 4 Issue 1
DOI: [To be assigned]

else
T = P
end if
if (p ≤ qmin && qk > p) do
qk = qk = -Δ
else if (p ≥ qmax && qk < p) do
qk = qk + Δ
else if ((qmin <p<qmax)&&(qk ≥ T))do
qk = min(255,qk+Δ
else
qk = min (0,qk - Δ)
end if
end for

PROPOSED ALGORITHM – DETERMINANT BASED

The algorithm for the determinant based fully automatic one scan adaptive image scaling algorithm is shown below:
Step 1: Load the input color image.
Step 2: Separate the three color components R, G, and B plane from the color image.
Step 3: Identify the eight neighbors of each pixel in each color components.
Step 4: Expand each pixel for n x n matrix named it as 'q' using determinant of weighted pixel.

For the zooming factor of 2,

$$q(1,1) = \begin{vmatrix} tl & t \\ l & p \end{vmatrix}$$
$$q(1,2) = \begin{vmatrix} t & tr \\ p & r \end{vmatrix}$$
$$q(2,1) = \begin{vmatrix} l & p \\ bl & b \end{vmatrix}$$
$$q(2,2) = \begin{vmatrix} p & r \\ b & br \end{vmatrix}$$

For the zooming factor of 3,

$$q(1,1) = \begin{vmatrix} tl & t \\ l & p \end{vmatrix}$$
$$q(1,2) = \begin{vmatrix} tl & t \\ l & p \end{vmatrix} + \begin{vmatrix} t & tr \\ p & r \end{vmatrix}$$
$$q(1,3) = \begin{vmatrix} t & tr \\ p & r \end{vmatrix}$$
$$q(2,1) = \begin{vmatrix} tl & t \\ l & p \end{vmatrix} + \begin{vmatrix} l & p \\ bl & b \end{vmatrix}$$
$$q(2,2) = \begin{vmatrix} tl & t \\ l & p \end{vmatrix} + \begin{vmatrix} t & tr \\ p & r \end{vmatrix} + \begin{vmatrix} l & p \\ bl & b \end{vmatrix} + \begin{vmatrix} p & r \\ b & br \end{vmatrix}$$
$$q(2,3) = \begin{vmatrix} t & tr \\ p & r \end{vmatrix} + \begin{vmatrix} p & r \\ b & br \end{vmatrix}$$
$$q(3,1) = \begin{vmatrix} l & p \\ bl & b \end{vmatrix}$$
$$q(3,2) = \begin{vmatrix} l & p \\ bl & b \end{vmatrix} + \begin{vmatrix} p & r \\ b & br \end{vmatrix}$$

HBRP PUBLICATION
Journal of Emerging Trends in Electrical Engineering
Volume 4 Issue 1
DOI: [To be assigned]

$$q(3,3) = \begin{vmatrix} p & r \\ b & br \end{vmatrix}$$

For the zooming factor of 4,

$$q(1,1) = \begin{vmatrix} tl & t \\ l & p \end{vmatrix}$$
$$q(1,2) = \begin{vmatrix} tl & t \\ l & p \end{vmatrix}$$
$$q(1,3) = \begin{vmatrix} t & tr \\ p & r \end{vmatrix}$$
$$q(1,4) = \begin{vmatrix} t & tr \\ p & r \end{vmatrix}$$
$$q(2,1) = \begin{vmatrix} tl & t \\ l & p \end{vmatrix}$$
$$q(2,2) = \begin{vmatrix} tl & t \\ l & p \end{vmatrix}$$
$$q(2,3) = \begin{vmatrix} t & tr \\ p & r \end{vmatrix}$$
$$q(2,4) = \begin{vmatrix} t & tr \\ p & r \end{vmatrix}$$
$$q(3,1) = \begin{vmatrix} l & p \\ bl & b \end{vmatrix}$$
$$q(3,2) = \begin{vmatrix} l & p \\ bl & b \end{vmatrix}$$
$$q(3,3) = \begin{vmatrix} p & r \\ b & br \end{vmatrix}$$
$$q(3,4) = \begin{vmatrix} p & r \\ b & br \end{vmatrix}$$
$$q(4,1) = \begin{vmatrix} l & p \\ bl & b \end{vmatrix}$$
$$q(4,2) = \begin{vmatrix} l & p \\ bl & b \end{vmatrix}$$
$$q(4,3) = \begin{vmatrix} p & r \\ b & br \end{vmatrix}$$
$$q(4,4) = \begin{vmatrix} p & r \\ b & br \end{vmatrix}$$

where p is the value of the pixel, and l, r, t, b, tl, tb, bl, and br are the neighbors of the pixel p.

Step 5: After computing q matrix find the maximum and minimum value in the q matrix.
Step 6: Find the value of Δ by using the formula

$$\Delta = \min_{q_k \neq p}\{|q_k - p|\}$$

Step 7: Find the value of T by using the formula

$$T = \begin{cases} q_{max} + q_{min} - p & if\ (q_{max} + q_{min})/2 \neq p \\ p & if\ (q_{max} + q_{min})/2 = p \end{cases}$$

Step 8: Find the final value of the pixel in q matrix is obtained by

$$q_k = \begin{cases} q_k - \Delta & if\ p \leq q_{min}\ for\ any\ q_k > p \\ q_k + p & if\ p \geq q_{max}\ for\ any\ q_k < p \\ \min(255, q_k + \Delta) & if\ q_{min} < p < q_{max}\ for\ any\ q_k \geq T \\ \min(0, q_k - \Delta) & if\ q_{min} < p < q_{max}\ for\ any\ q_k < T \end{cases}$$

Step 9: Finally calculate the PSNR and SSIM between the input and zoomed algorithm.

HBRP PUBLICATION
Journal of Emerging Trends in Electrical Engineering
Volume 4 Issue 1
DOI: [To be assigned]

$$PSNR = 20\log_{10}\left(\frac{255}{\sqrt{MSE}}\right)$$

where MSE is the mean square error.

$$SSIM = \frac{(2\mu_v\mu_w + c_1)(2cov_{vw} + c_2)}{(\mu_v^2 + \mu_w^2 + c_1)(\sigma_v^2 + \sigma_w^2 + c_2)}$$

μ_v is the mean of the input image
μ_w is the mean of the zoomed image
σ_v^2 is the variance of the input image
σ_w^2 is the variance of the zoomed image
$c_1 = (k_1L)^2$ and $c_2 = (k_2L)^2$
$k_1 = 0.01$; $k_2 = 0.03$ and $L = 255$

The Pseudocode for the determinant based fully automatic one scan adaptive image scaling algorithm is shown below:

P ← Input image
Q ← Output image
N ← zooming factor
for each pixel in p do
tl=p(i-1,j-1,k);
t=p(i-1,j,k);
tr=p(i-1,j+1,k);
l=p(i,j-1,k);
p=p(i,j,k);
r=p(i,j+1,k);
bl=p(i+1,j-1,k);
b=p(i+1,j,k);
br=p(i+1,j+1,k);
end for
for each pixel in p do
    for i = 1 to n do
if N=2 do

$$q(1,1) = \begin{vmatrix} tl & t \\ l & p \end{vmatrix}$$

$$q(1,2) = \begin{vmatrix} t & tr \\ p & r \end{vmatrix}$$

$$q(2,1) = \begin{vmatrix} l & p \\ bl & b \end{vmatrix}$$

$$q(2,2) = \begin{vmatrix} p & r \\ b & br \end{vmatrix}$$

elseif N=3 do

$$q(1,1) = \begin{vmatrix} tl & t \\ l & p \end{vmatrix}$$

$$q(1,2) = \begin{vmatrix} tl & t \\ l & p \end{vmatrix} + \begin{vmatrix} t & tr \\ p & r \end{vmatrix}$$

$$q(1,3) = \begin{vmatrix} t & tr \\ p & r \end{vmatrix}$$

$$q(2,1) = \begin{vmatrix} tl & t \\ l & p \end{vmatrix} + \begin{vmatrix} l & p \\ bl & b \end{vmatrix}$$

Journal of Emerging Trends in Electrical Engineering
Volume 4 Issue 1
DOI: [To be assigned]

$q(2,2) = \begin{vmatrix} tl & t \\ l & p \end{vmatrix} + \begin{vmatrix} t & tr \\ p & r \end{vmatrix} + \begin{vmatrix} l & p \\ bl & b \end{vmatrix} + \begin{vmatrix} p & r \\ b & br \end{vmatrix}$

$q(2,3) = \begin{vmatrix} t & tr \\ p & r \end{vmatrix} + \begin{vmatrix} p & r \\ b & br \end{vmatrix}$

$q(3,1) = \begin{vmatrix} l & p \\ bl & b \end{vmatrix}$

$q(3,2) = \begin{vmatrix} l & p \\ bl & b \end{vmatrix} + \begin{vmatrix} p & r \\ b & br \end{vmatrix}$

$q(3,3) = \begin{vmatrix} p & r \\ b & br \end{vmatrix}$

else // N=4

$q(1,1) = \begin{vmatrix} tl & t \\ l & p \end{vmatrix}$

$q(1,2) = \begin{vmatrix} tl & t \\ l & p \end{vmatrix}$

$q(1,3) = \begin{vmatrix} t & tr \\ p & r \end{vmatrix}$

$q(1,4) = \begin{vmatrix} t & tr \\ p & r \end{vmatrix}$

$q(2,1) = \begin{vmatrix} tl & t \\ l & p \end{vmatrix}$

$q(2,2) = \begin{vmatrix} tl & t \\ l & p \end{vmatrix}$

$q(2,3) = \begin{vmatrix} t & tr \\ p & r \end{vmatrix}$

$q(2,4) = \begin{vmatrix} t & tr \\ p & r \end{vmatrix}$

$q(3,1) = \begin{vmatrix} l & p \\ bl & b \end{vmatrix}$

$q(3,2) = \begin{vmatrix} l & p \\ bl & b \end{vmatrix}$

$q(3,3) = \begin{vmatrix} p & r \\ b & br \end{vmatrix}$

$q(3,4) = \begin{vmatrix} p & r \\ b & br \end{vmatrix}$

$q(4,1) = \begin{vmatrix} l & p \\ bl & b \end{vmatrix}$

$q(4,2) = \begin{vmatrix} l & p \\ bl & b \end{vmatrix}$

$q(4,3) = \begin{vmatrix} p & r \\ b & br \end{vmatrix}$

$q(4,4) = \begin{vmatrix} p & r \\ b & br \end{vmatrix}$

end if
end for
if ($qk \neq p$) do
$\Delta = \min (|qk - p|)$
end if
if $(q\max + q\min)/2 \neq p$ do
$t = q\max + q\min - p$
else
T = P
end if

Journal of Emerging Trends in Electrical Engineering
Volume 4 Issue 1
DOI: [To be assigned]

```
if ( p ≤ qmin && qk > p) do
qk = qk = -Δ
else if (p ≥ qmax && qk < p) do
qk = qk + Δ
else if ((qmin <p<qmax)&&(qk ≥ T))do
qk = min(255,qk+Δ
else
qk = min (0,qk - Δ)
end if
end for
```

PROPOSED ALGORITHM – WEIGHTED DETERMINANT BASED

The algorithm for the weighted determinant based fully automatic one scan adaptive image scaling algorithm is shown below:

Step 1: Load the input color image.

Step 2: Separate the three color components R, G, and B plane from the color image.

Step 3: Identify the eight neighbors of each pixel in each color components.

Step 4: Expand each pixel for n x n matrix named it as ‘q’ using determinant of weighted pixel.

For the zooming factor of 2,

$$q(1,1) = \begin{vmatrix} tl & 2t \\ 2l & 4p \end{vmatrix}$$

$$q(1,2) = \begin{vmatrix} 2t & tr \\ 4p & 2r \end{vmatrix}$$

$$q(2,1) = \begin{vmatrix} 2l & 4p \\ bl & 2b \end{vmatrix}$$

$$q(2,2) = \begin{vmatrix} 4p & 2r \\ 2b & br \end{vmatrix}$$

For the zooming factor of 3,

$$q(1,1) = \begin{vmatrix} 4tl & 6t \\ 6l & 9p \end{vmatrix}$$

$$q(1,2) = \begin{vmatrix} 2tl & 6t \\ 3l & 9p \end{vmatrix} + \begin{vmatrix} 6t & 2tr \\ 9p & 3r \end{vmatrix}$$

$$q(1,3) = \begin{vmatrix} 6t & 4tr \\ 9p & 6r \end{vmatrix}$$

$$q(2,1) = \begin{vmatrix} 2tl & 3t \\ 6l & 9p \end{vmatrix} + \begin{vmatrix} 6l & 9p \\ 2bl & 3b \end{vmatrix}$$

$$q(2,2) = \begin{vmatrix} tl & 3t \\ 3l & 9p \end{vmatrix} + \begin{vmatrix} 3t & tr \\ 9p & 3r \end{vmatrix} + \begin{vmatrix} 3l & 9p \\ bl & 3b \end{vmatrix} + \begin{vmatrix} 9p & 3r \\ 3b & br \end{vmatrix}$$

$$q(2,3) = \begin{vmatrix} 3t & 2tr \\ 9p & 6r \end{vmatrix} + \begin{vmatrix} 9p & 6r \\ 3b & 2br \end{vmatrix}$$

$$q(3,1) = \begin{vmatrix} 6l & 9p \\ 4bl & 6b \end{vmatrix}$$

$$q(3,2) = \begin{vmatrix} 3l & 9p \\ 2bl & 6b \end{vmatrix} + \begin{vmatrix} 9p & 3r \\ 6b & 2br \end{vmatrix}$$

$$q(3,3) = \begin{vmatrix} 9p & 6r \\ 6b & 4br \end{vmatrix}$$

Journal of Emerging Trends in Electrical Engineering
Volume 4 Issue 1
DOI: [To be assigned]

For the zooming factor of 4,

$$q(1,1) = \begin{vmatrix} 4tl & 6t \\ 6l & 9p \end{vmatrix}$$
$$q(1,2) = \begin{vmatrix} 2tl & 8t \\ 3l & 12p \end{vmatrix}$$
$$q(1,3) = \begin{vmatrix} 8t & 2tr \\ 12p & 3r \end{vmatrix}$$
$$q(1,4) = \begin{vmatrix} 6t & 4tr \\ 12p & 6r \end{vmatrix}$$
$$q(2,1) = \begin{vmatrix} 2tl & 3t \\ 8l & 12p \end{vmatrix}$$
$$q(2,2) = \begin{vmatrix} tl & 4t \\ 4l & 16p \end{vmatrix}$$
$$q(2,3) = \begin{vmatrix} 4t & tr \\ 16p & 4r \end{vmatrix}$$
$$q(2,4) = \begin{vmatrix} 3t & 2tr \\ 12p & 8r \end{vmatrix}$$
$$q(3,1) = \begin{vmatrix} 8l & 12p \\ 2bl & 3b \end{vmatrix}$$
$$q(3,2) = \begin{vmatrix} 4l & 16p \\ bl & 4b \end{vmatrix}$$
$$q(3,3) = \begin{vmatrix} 16p & 4r \\ 4b & br \end{vmatrix}$$
$$q(3,4) = \begin{vmatrix} 12p & 8r \\ 3b & 2br \end{vmatrix}$$
$$q(4,1) = \begin{vmatrix} 6l & 9p \\ 4bl & 6b \end{vmatrix}$$
$$q(4,2) = \begin{vmatrix} 3l & 12p \\ 2bl & 8b \end{vmatrix}$$
$$q(4,3) = \begin{vmatrix} 12p & 3r \\ 8b & 2br \end{vmatrix}$$
$$q(4,4) = \begin{vmatrix} 9p & 6r \\ 6b & 4br \end{vmatrix}$$

where p is the value of the pixel, and l, r, t, b, tl, tb, bl, and br are the neighbors of the pixel p.

Step 5: After computing q matrix find the maximum and minimum value in the q matrix.
Step 6: Find the value of Δ by using the formula

$$\Delta = \min_{q_k \neq p}\{|q_k - p|\}$$

Step 7: Find the value of T by using the formula

$$T = \begin{cases} q_{max} + q_{min} - p & if\ (q_{max} + q_{min})/2 \neq p \\ p & if\ (q_{max} + q_{min})/2 = p \end{cases}$$

Step 8: Find the final value of the pixel in q matrix is obtained by

$$q_k = \begin{cases} q_k - \Delta & if\ p \leq q_{min}\ for\ any\ q_k > p \\ q_k + p & if\ p \geq q_{max}\ for\ any\ q_k < p \\ \min(255, q_k + \Delta) & if\ q_{min} < p < q_{max}\ for\ any\ q_k \geq T \\ \min(0, q_k - \Delta) & if\ q_{min} < p < q_{max}\ for\ any\ q_k < T \end{cases}$$

Step 9: Finally calculate the PSNR and SSIM between the input and zoomed algorithm.

$$PSNR = 20 \log_{10}\left(\frac{255}{\sqrt{MSE}}\right)$$

HBRP PUBLICATION
Journal of Emerging Trends in Electrical Engineering
Volume 4 Issue 1
DOI: [To be assigned]

where MSE is the mean square error.

$$SSIM = \frac{(2\mu_v\mu_w + c_1)(2cov_{vw} + c_2)}{(\mu_v^2 + \mu_w^2 + c_1)(\sigma_v^2 + \sigma_w^2 + c_2)}$$

μ_v is the mean of the input image
μ_w is the mean of the zoomed image
σ_v^2 is the variance of the input image
σ_w^2 is the variance of the zoomed image
$c_1 = (k_1L)^2$ and $c_2 = (k_2L)^2$
$k_1 = 0.01$; $k_2 = 0.03$ and $L = 255$

The Pseudocode for the weighted determinant based fully automatic one scan adaptive image scaling algorithm is shown below:

P ← Input image
Q ← Output image
N ← zooming factor
for each pixel in p do
    tl=p(i-1,j-1,k);
t=p(i-1,j,k);
tr=p(i-1,j+1,k);
l=p(i,j-1,k);
p=p(i,j,k);
r=p(i,j+1,k);
bl=p(i+1,j-1,k);
b=p(i+1,j,k);
br=p(i+1,j+1,k);
end for
for each pixel in p do
    for i = 1 to n do
if N=2 do

$$q(1,1) = \begin{vmatrix} tl & 2t \\ 2l & 4p \end{vmatrix}$$

$$q(1,2) = \begin{vmatrix} 2t & tr \\ 4p & 2r \end{vmatrix}$$

$$q(2,1) = \begin{vmatrix} 2l & 4p \\ bl & 2b \end{vmatrix}$$

$$q(2,2) = \begin{vmatrix} 4p & 2r \\ 2b & br \end{vmatrix}$$

elseif N=3 do

$$q(1,1) = \begin{vmatrix} 4tl & 6t \\ 6l & 9p \end{vmatrix}$$

$$q(1,2) = \begin{vmatrix} 2tl & 6t \\ 3l & 9p \end{vmatrix} + \begin{vmatrix} 6t & 2tr \\ 9p & 3r \end{vmatrix}$$

$$q(1,3) = \begin{vmatrix} 6t & 4tr \\ 9p & 6r \end{vmatrix}$$

$$q(2,1) = \begin{vmatrix} 2tl & 3t \\ 6l & 9p \end{vmatrix} + \begin{vmatrix} 6l & 9p \\ 2bl & 3b \end{vmatrix}$$

$$q(2,2) = \begin{vmatrix} tl & 3t \\ 3l & 9p \end{vmatrix} + \begin{vmatrix} 3t & tr \\ 9p & 3r \end{vmatrix} + \begin{vmatrix} 3l & 9p \\ bl & 3b \end{vmatrix} + \begin{vmatrix} 9p & 3r \\ 3b & br \end{vmatrix}$$

Journal of Emerging Trends in Electrical Engineering
Volume 4 Issue 1
DOI: [To be assigned]

$q(2,3) = \begin{vmatrix} 3t & 2tr \\ 9p & 6r \end{vmatrix} + \begin{vmatrix} 9p & 6r \\ 3b & 2br \end{vmatrix}$

$q(3,1) = \begin{vmatrix} 6l & 9p \\ 4bl & 6b \end{vmatrix}$

$q(3,2) = \begin{vmatrix} 3l & 9p \\ 2bl & 6b \end{vmatrix} + \begin{vmatrix} 9p & 3r \\ 6b & 2br \end{vmatrix}$

$q(3,3) = \begin{vmatrix} 9p & 6r \\ 6b & 4br \end{vmatrix}$

else // N=4

$q(1,1) = \begin{vmatrix} 4tl & 6t \\ 6l & 9p \end{vmatrix}$

$q(1,2) = \begin{vmatrix} 2tl & 8t \\ 3l & 12p \end{vmatrix}$

$q(1,3) = \begin{vmatrix} 8t & 2tr \\ 12p & 3r \end{vmatrix}$

$q(1,4) = \begin{vmatrix} 6t & 4tr \\ 12p & 6r \end{vmatrix}$

$q(2,1) = \begin{vmatrix} 2tl & 3t \\ 8l & 12p \end{vmatrix}$

$q(2,2) = \begin{vmatrix} tl & 4t \\ 4l & 16p \end{vmatrix}$

$q(2,3) = \begin{vmatrix} 4t & tr \\ 16p & 4r \end{vmatrix}$

$q(2,4) = \begin{vmatrix} 3t & 2tr \\ 12p & 8r \end{vmatrix}$

$q(3,1) = \begin{vmatrix} 8l & 12p \\ 2bl & 3b \end{vmatrix}$

$q(3,2) = \begin{vmatrix} 4l & 16p \\ bl & 4b \end{vmatrix}$

$q(3,3) = \begin{vmatrix} 16p & 4r \\ 4b & br \end{vmatrix}$

$q(3,4) = \begin{vmatrix} 12p & 8r \\ 3b & 2br \end{vmatrix}$

$q(4,1) = \begin{vmatrix} 6l & 9p \\ 4bl & 6b \end{vmatrix}$

$q(4,2) = \begin{vmatrix} 3l & 12p \\ 2bl & 8b \end{vmatrix}$

$q(4,3) = \begin{vmatrix} 12p & 3r \\ 8b & 2br \end{vmatrix}$

$q(4,4) = \begin{vmatrix} 9p & 6r \\ 6b & 4br \end{vmatrix}$

end if

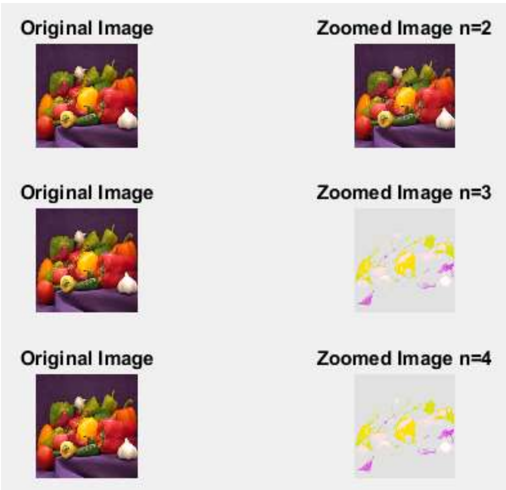
end for
if (qk ≠ p) do
Δ = min (|qk – p|)
end if
if (q max + qmin)/2 ≠ p do
t = q max + qmin – p
else
T = P
end if

HBRP PUBLICATION
Journal of Emerging Trends in Electrical Engineering
Volume 4 Issue 1
DOI: [To be assigned]

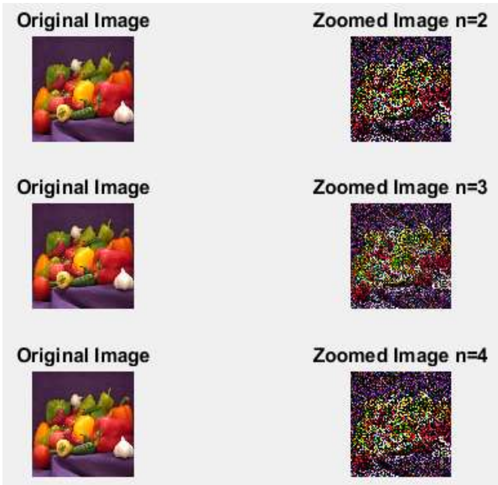
```
        if ( p ≤ qmin && qk > p) do
                qk = qk = -Δ
        else if (p ≥ qmax && qk < p) do
                qk = qk + Δ
        else if ((qmin <p<qmax)&&(qk ≥ T))do
                qk = min(255,qk+Δ
else
                qk = min (0,qk - Δ)

end if
end for
```

RESULTS AND DISCUSSION

To evaluate the result we consider the MATLAB default image namely wpeppers.jpg.Figure 1 (a), (b) and (c) shows the input image and zoomed image with zooming factor of 2, 3, and 4 for the existing weighted sum, and proposed determinant and weighted determinant based fully automatic one scan adaptive image scaling algorithm respectively for wpeppers.jpg.



(a) Weighted Sum Based



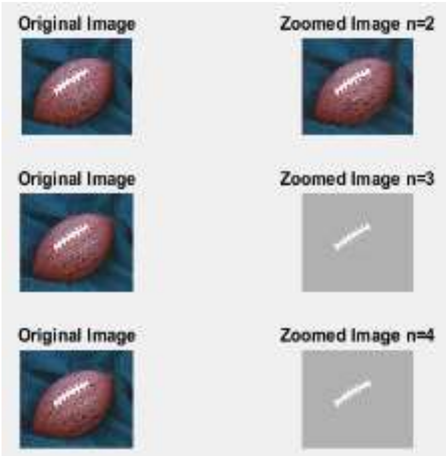
(b) Determinant Based

(c) Weighted Determinant Based

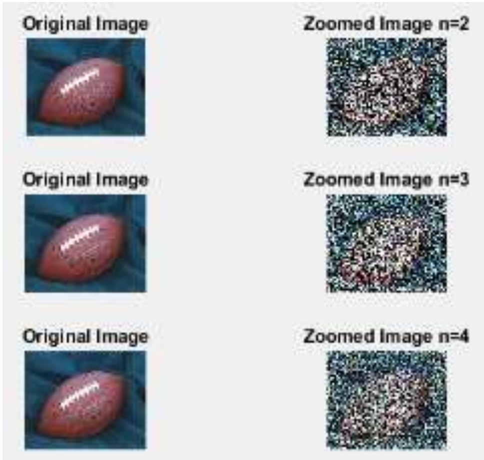
***Fig. 1:** Input and Output Image for Existing and Proposed Method for Wpeppers.jpg*

Journal of Emerging Trends in Electrical Engineering
Volume 4 Issue 1
DOI: [To be assigned]

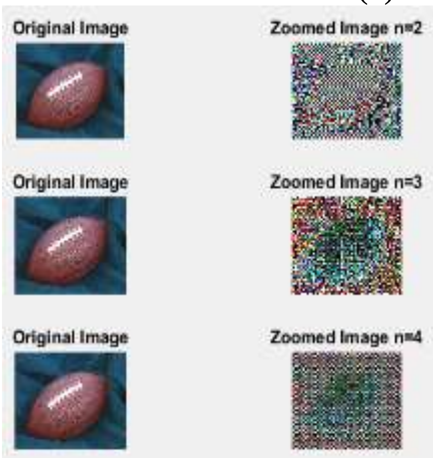
Figure 2 (a), (b) and (c) shows the input image and zoomed image with zooming factor of 2, 3, and 4 for the existing weighted sum, and proposed determinant and weighted determinant based fully automatic one scan adaptive image scaling algorithm respectively for football.jpg.



(a)Weighted Sum Based



(b) Determinant Based



(c) Weighted Determinant Based

Fig. 2: *Input and Output Image for Existing and Proposed Method for Football.*

Table 1 shows the comparison of exiting weighted sum based and proposed determinant and weighted determinant based algorithm in terms of PSNR and SSIM.

CONCLUSION

From the table 5.1, it is concluded that the PSNR and SSIM value of the proposed determinant and weighted determinant based image scaling algorithm is worst compared to the existing weighted sum based image scaling algorithm. It is evident by seeing the figure 1 and 2 that the image quality is poor in case of proposed system compared to existing systems. In future, we will analyze why this method is not efficient through the theoretic calculation and also will able to explain how this problem can be rectified. Also we can see from figure 1 and 2 that the information from the zoomed image with zooming factor 3 and 4 is known in case of proposed system than existing system, i.e., in existing system, information and noise both are less for zooming factor 3 and 4 and in proposed system, information and noise both are high for zooming factor 3 and 4.

Journal of Emerging Trends in Electrical Engineering
Volume 4 Issue 1
DOI: [To be assigned]

Table 1: *Comparison Table.*

Image	Zooming Factor	Carlo Arcelli *et al* Weighted Sum based		Proposed Determinant based		Proposed Weighted Determinant based	
		PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
wpeppers. jpg	2	30.0596	$1.9277 * 10^{-19}$	-2.3702	$2.5836 * 10^{-36}$	-0.8106	$2.4314 * 10^{-36}$
	3	4.5864	$4.5072 * 10^{-23}$	-1.9219	$1.0881 * 10^{-37}$	-10.7206	$-5.1015 * 10^{-41}$
	4	4.5882	$8.1732 * 10^{-23}$	-2.0859	$2.5661 * 10^{-37}$	-1.2910	$-4.0028 * 10^{-39}$
football.jpg	2	32.5538	$1.3484 * 10^{-21}$	-5.5334	$-6.8174 * 10^{-39}$	-2.9893	$1.1175 * 10^{-39}$
	3	7.5004	$6.1611 * 10^{-22}$	-7.2121	$6.8290 * 10^{-40}$	-17.1157	$2.1201 * 10^{-44}$
	4	7.5005	$1.0884 * 10^{-21}$	-4.4485	$-3.4114 * 10^{-39}$	-1.1430	$-1.1410 * 10^{-39}$

REFERENCE

1. Arcelli, C., Brancati, N., Frucci, M., Ramella, G., & Di Baja, G. S. (2011). A fully automatic one-scan adaptive zooming algorithm for color images. *Signal Processing, 91*(1), 61-71.
2. Li, X., & Orchard, M. T. (2001). New edge-directed interpolation. *IEEE transactions on image processing, 10*(10), 1521-1527.
3. Reddy, K. S., & Reddy, D. K. R. L. (2013). Enlargement of image based upon Interpolation Techniques. *International Journal of Advanced Research in Computer and Communication Engineering, 2*(12), 4631.

Cite this article as: [To be assigned]

BIBILIOGRAPHY

Sivarajan Rajendran has received B.E. degree in the field of Electronics and Communication Engineering in 2010 from Adhiparasakthi Engineering College (Affiliated to Anna University) and M.E. degree in the field of Communication Systems in 2012 from Sree Sastha Institute of Engineering and Technology (Affiliated to Anna University). He started his teaching carrier in the year 2013 at the Adhiparasakthi Engineering College, Melmaruvathur. His area of interest is Signal and Image Processing,

Journal of Emerging Trends in Electrical Engineering
Volume 4 Issue 1
DOI: [To be assigned]

Electromagnetism and VLSI design. He published 8 papers in International Journal, 3 papers in National Journal, 6 papers in International conference and 4 papers in National Conference. He has completed 2 NPTEL online Certification program in the title of An Introduction to Information Theory (Jan-Mar 2021) and An Introduction to Coding Theory (Sep-Nov 2020). He has also attended various Faculty Development Program, Workshops, and Webinars. He has also published a book in the title "Fundamentals of Analog and Digital Signal Processing" in Dipti Press Pvt. Ltd. in April 2019.

Selvam. K has received B.E. degree in the field of Electronics and Communication Engineering in 2017 from Adhiparasakthi Engineering College (Affiliated to Anna University). He is currently pursuing PG course in the field of VLSI Design.

Dr. Nagarajan Velmurugan has received B.E. degree in the field of Electronics and Communication Engineering in 1999 from VRS Engineering College (Affiliated to Madras University), M.E. degree in the field of ECE in 2002 from Pondicherry Engineering College and Ph.D. degree in the area of wireless communication in 2010 from Pondicherry University. He is a celebrated member of Professional bodies such as Institute of Electronics and Telecommunication Engineering (IETE) in New Delhi, and senior member of IEEE. He has published more than 150 publications in international journals and conferences. Under his guidance 6 scholars were awarded Ph.D and 12 scholars are pursuing Ph.D. He has received best teaching award for two times and received IEEE best paper awards for 3 times. He has visited various foreign countries namely Singapore, Malaysia, Thailand, South Korea, Sharaja and South Africa.

Vedasundaravinayagam Palanivel has received B.E. degree in the field of Electronics and Communication Engineering in 2007 from Adhiyaman Engineering College (Affiliated to Anna University) and M.E. degree in the field of Applied Electronics in 2011 from Tiruvalluvar College of Engineering and Technology (Affiliated to Anna University). He started his teaching carrier in the year 2008 as a Lecturer in Lord Venkateshwara Engineering College, Wallajabad. He is currently working as Assistant Professor in Adhiparasakthi Engineering College, Melmaruvathur. He has a 1 year of industrial experience as Instructor in I-Software Integration Technologies Pvt. Ltd., Bangalore. His area of interest is Microprocessor and Microcontroller and Embedded Systems. He published 3 papers in International Journal, 3 papers in International conference and 1 paper in National Conference.