Volume 2, Issue 7, July 2012 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

# CA Based Moore Filter in SEO: To Enhance Image Ranking

**Arnab Mitra, Sourav Samaddar**
*Dept. of CSE, Adamas Institute of Technology,*
*West Bengal, India-700126*

***Abstract— In recent days, web information retrieval through various search engines plays an important role in quick access of knowledge. This information retrieval includes both textual and image extraction. The search engine serves the user query result in an indexed way which introduces the proper ranking of the relevant results. It is desirable that higher quality information should have higher priority i.e. higher ranking in that particular indexing. In this research work, we have put forward our emphasis on the improvement of image ranking in the area of search engine optimization (SEO) based on the quality of an image. The quality improvement for image refers to the "peeper and salt" noise reduction from the image by Cellular Automata (CA) based image processing techniques. The experimental result signifies that this proposed methodology is capable of providing an improved image ranking.***

***Keywords— Search Engine Optimization (SEO), Information Extraction, Page-Ranking, Image-Ranking, Image Processing, Noise Filtering, Peeper & Salt Noise Removal, Cellular Automata (CA)***

## I. Introduction

In computer graphics, a raster graphics image, or bitmap, is a data structure representing a generally rectangular grid of pixels, or points of colour, viewable via a monitor, paper, or other display medium. Raster images are stored in image files with varying formats. A bitmap correspond bit-for-bit with an image displayed on a screen, generally in the same format used for storage in the display's video memory, or maybe as a device-independent bitmap. A bitmap is technically characterized by the width and height of the image in pixels and by the number of bits per pixel (a colour depth, which determines the number of colours it can represent).

### *A. Type of Images*

All electronic art images are divided into one of two core types, raster images (also known as 'bitmap') and vector images. In a nutshell raster images are composed of connected dots and vectors are images composed of connected lines. Raster images are created through the process of scanning source artwork or "painting" with a photo editing or paint program such as Corel Photo PAINT or Adobe Photoshop. Vector images are created through the process of drawing with vector illustration programs such as CorelDraw or Adobe Illustrator. The word "vector" is a synonym for line. Vector images can also be created through the process of conversion from a raster image by using a raster to vector conversion program such as Corel TRACE or Euro VECTOR.

### *B. Cellular Automata*

A **cellular automaton** (pl. **cellular automata**, abbrev. **CA**) is a discrete model studied in computability theory, mathematics, physics, complexity science, theoretical biology and microstructure modelling. It consists of a regular grid of cells, each in one of a finite number of states, such as "On" and "Off" (in contrast to a coupled map lattice) as shown in Figure 1. The grid can be in any finite number of dimensions as shown in Figure 2. For each cell, a set of cells called its neighbourhood (usually including the cell itself) is defined relative to the specified cell. A new generation is created (advancing *t* by 1), according to some fixed rule (generally, a mathematical function) that determines the new state of each cell in terms of the current state of the cell and the states of the cells in its neighbourhood. The simplest nontrivial CA would be one dimensional, with two possible states per cell, and a cell's neighbours defined to be the adjacent cells on either side of it. A cell and its two neighbours form a neighbourhood of 3 cells, so there are 23=8 possible patterns for a neighbourhood. There are then 28=256 possible rules. These 256 CAs are generally referred to by their Wolfram code, a standard naming convention invented by Wolfram, which gives each rule a number from 0 to 255. A number of papers have analysed and compared these 256 CAs, either individually or collectively. The rule 30 and rule 110 CAs are particularly interesting. The grid of cells of CA, each of which can be in only one of a finite number of possible states [14].

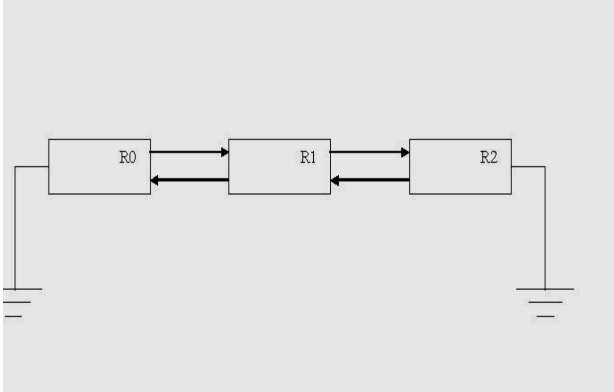


**Figure 1: Implementation of 3 Cell null boundaries CA**

The state of a cell is determined by the previous states of a surrounding neighbourhood of cells and is updated synchronously in discrete time steps. One of the advantages of CAs is that, although each cell generally only contains a few simple rules, the combination of a matrix of cells with their local interaction leads to more sophisticated emergent global behaviour. Following Figure 1 and Figure 2 shows a basic diagram of Cellular Automata.

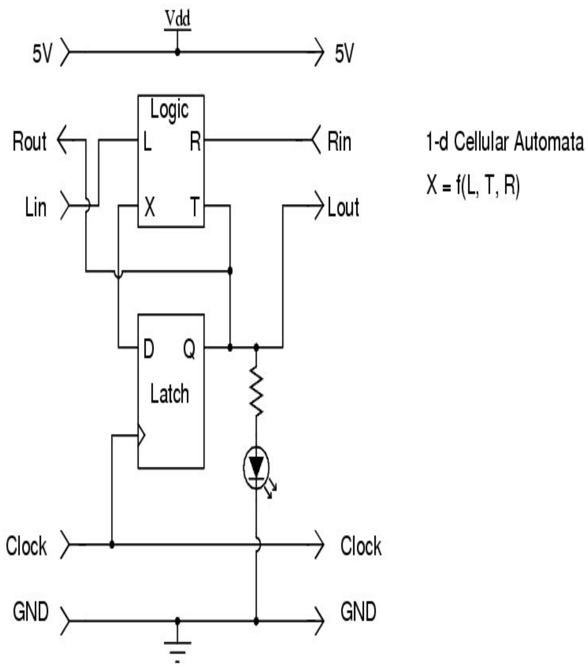


**Figure 2: Basic hardware implementation for CA structure**

That is, although each cell has an extremely limited view of the system (just its immediate neighbours), localized information is propagated at each time step, enabling more global characteristics of the overall CA system. The cellular automaton uses the set {0,1} for its two states. The size of the rule (as a decimal number) is found by the following equation where n is the size of the cell neighbourhood: rule size=$2^n$.This means that the rule space grows very exponentially as the neighbourhood size increases, examples of the rule space sizes are given below in Table 1:

**Table 1: CA rule size and bit relation**

| Number of Cell=n | Decimal Rule | Size Rule Bit Rate |
|---|---|---|
| 2 | 16 | 4 |
| 3 | 256 | 8 |
| 4 | 65536 | 16 |
| 5 | 4294967296 | 32 |
| 6 | 1.86EE19 | 64 |
| 7 | 3.40EE38 | 128 |
| 8 | 1.16EE77 | 256 |
| 9 | 1.34EE154 | 512 |

By mathematics the 2-dimensional cellular automata computation consists of a regular lattice of sites. Each site takes on –possible values, and is updated in discrete time steps according to a rule that depends on the value of sites in some neighbourhood around it. The value of a site at position in a one-dimensional cellular automaton with a rule that depends only on nearest neighbours thus evolves according to some mathematical equations.

*C. Page-Rank*

Page-Ranking is a link analysis algorithm and is being used by the several popular search engines all over the world, that assigns a numerical weighting to each element of a hyperlinked set of documents that includes both the textual and images, such as the World Wide Web, with the purpose of measuring its relative importance within the set. The algorithm may be applied to any collection of entities with reciprocal quotations and references. The numerical weight that it assigns to any given element *E* is referred to as the Page-Rank of E and denoted by

A Page-Rank results from a mathematical algorithm based on the graph, the webgraph, created by all World Wide Web pages as nodes and hyperlinks as edges, taking into consideration authority hubs such as cnn.com or usa.gov. The rank value indicates an importance of a particular page. A hyperlink to a page counts as a vote of support. The Page-Rank of a page is defined recursively and depends on the number and Page-Rank metric of all pages that link to it as incoming links. A page that is linked to by many pages with high Page-Rank receives a high rank itself. If there are no links to a web page there is no support for that page.

Numerous academic papers concerning Page-Rank have been published since Page and Brin's original paper. In practice, the Page-Rank concept has proven to be vulnerable to manipulation, and extensive research has been devoted to identifying falsely inflated Page-Rank and ways to ignore links from documents with falsely inflated Page-Rank.

Other link-based ranking algorithms for Web pages include the HITS algorithm invented by Jon Kleinberg (used by Teoma and now Ask.com), the IBM CLEVER project, and the TrustRank algorithm.

Page-Rank is a probability distribution used to represent the likelihood that a person randomly clicking on links will arrive at any particular page. Page-Rank can be calculated for collections of documents of any size. Today, typical web documents not only include text information only, but a plenty of images and other multimedia files are being contained. A webpage containing n-number of pictures/images are considered as to be a collection of n+1 number of files. It is assumed in several research papers that the distribution is evenly divided among all documents in the collection at the beginning of the computational process. The Page-Rank computations require several passes, called "iterations", through the collection to adjust approximate Page-Rank values to more closely reflect the theoretical true value.

A probability is expressed as a numeric value between 0 and 1. A 0.5 probability is commonly expressed as a "50% chance" of something happening. Hence, a Page-Rank of 0.5 means there is a 50% chance that a person clicking on a random link will be directed to the document with the 0.5 Page-Rank.

Assume a small universe of some web pages: **A**, **B**, **C, D, E** and F. The initial approximation of Page-Rank would be evenly divided between all these documents. Hence, each document would begin with an estimated Page-Rank of 0.25. Following Figure 3 shows a schematic representation of linked structure of pages [1].

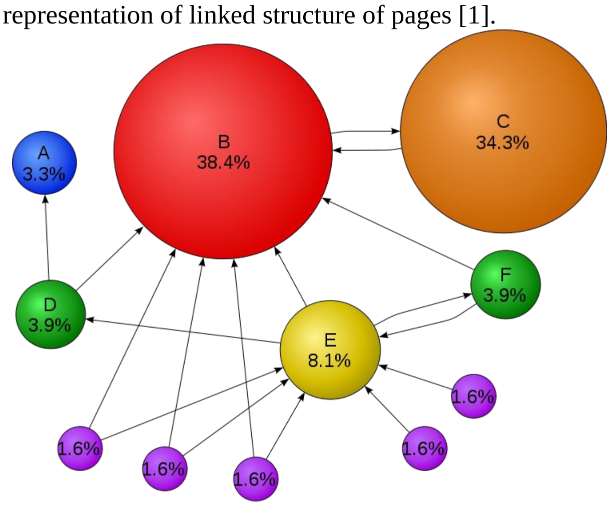


**Figure 3. Linked structure of web pages classified in google's example**

Mathematical Page-Ranks (out of 100) for a simple network (Page-Ranks reported by Google are rescaled logarithmically). Page C has a higher Page-Rank than Page E, even though it has fewer links to it; the link it has is of a much higher value. A web surfer who chooses a random link on every page (but with 15% likelihood jumps to a random page on the whole web) is going to be on Page E for 8.1% of the time. (The 15% likelihood of jumping to an arbitrary page corresponds to a damping factor of 85%.) Without damping, all web surfers would eventually end up on Pages A, B, or C, and all other pages would have Page-Rank zero. Page A is assumed to link to all pages in the web, because it has no outgoing links.

In other words, the Page-Rank conferred by an outbound link is equal to the document's own Page-Rank score divided by the normalized number of outbound links **L ()** (it is assumed that links to specific URLs only count once per document). [1]

Image enhancement is the processing of images to improve their appearance to human viewers or to enhance the performance of other image processing system [2]. In most applications involving images or image processing one of the most common problems is the presence of noise. The objective of image enhancement (for example improving image quality, intelligibility, visual appearance) is dependent of application context. An image enhancement algorithm that performs well for one class of images may not perform as well for other classes. Classically, image enhancement is formulated in either spatial or transform (basically the Fourier transform) domains. One of the most used spatial domain techniques is that of the so-called convolution masks. Such an example may be the Gaussian Filter. In transform domains, perhaps the most famous technique is the Wiener Filter. These procedures are generally based on calculations of directional derivatives that result in computationally intensive tasks or previous knowledge of the image nature [3]. However, these last requirements limit the applicability of the process.

Cellular automata were introduced by Von Neumann [4]. They have been progressively used to model a great variety of dynamical systems in different application domains. A cellular automaton is basically a computer algorithm that is discrete in space and time and operates on a lattice of sites (in our case, pixels). Using some predefined mathematical rule, CA can be used to model for filtering purpose of digital images.

A digital image is a bi-dimensional array of n x n pixels. Each pixel can be characterized by the triplet (i; j; k) where (i; j) represents its position in the array and k the associated color. The image may be then considered as a particular configuration state of a cellular automaton that has as cellular space the n x n array defined by the image. Each site in the array corresponds to a pixel.

Rest of the paper is organized as follows: Section 2 briefs related work; Section 3 describes proposed work; Experimental results are shown in Section 4; Conclusion is depicted in Section 5 and Future scope is mentioned in Section 6.

## II. RELATED WORK

Several attempts have been established to reduce noise from images. Among them Gaussian noise filter is quite popular [5]. Besides the Gaussian filter, previously developed CA Model for Filtering Digital Image refers a digital image is a bi dimensional array of n X n pixels. Each pixel can be characterized by the triplet (i, j, k) where (i, j) represents its position in the array and k the associated color. The image may be then considered as a particular configuration state of a cellular automaton that has as cellular space the n X n array defined by the image. Each site in the array corresponds to a pixel [8].
Several works have been carried out by researchers for Page Rank Calculation. Wikipedia enlists some the major attempts [3]. Particularly for Image rank calculation some research shows a better way to improve image rank based on different factors [5-7]. In Gaussian Noise filtration related works, A new faster and efficient algorithm was established by V.R.Vijaykumar et al which is capable in removing Gaussian noise with less computational complexity. But its physical implementation is costly [8-13].

In our previous research we have shown that Moore neighborhood based 2-dimensional CA can remove noise with lesser cost. This CA based model can be easily implemented in hardware as it is flip-flop based. The noise reduction quality is also quite impressive [14].

## III. PROPOSED METHODOLOGY

In this work, we have proposed a dynamical searching system which will provide better image-ranking while reducing the peeper and salt noise successfully if it presents in any image. Instead of providing direct image-ranking, this system should first check whether any of the required images contains the peeper and salt noise or not. If it finds it, then an CA based noise filtration technique is used to enhance the quality of the image. In our approach, we have chosen Moore neighbourhood instead of Von-Neumann neighbourhood. Moore neighbourhood consists of 8 neighbours so that the detection of the noise is performed in a better way. For this requirement we have modified the existing Von-Neumann neighbourhood algorithm. Our established methodology is described as following Figure 4 [14]. Based on the proposed system's flowchart in Figure 4, following Algorithm 1 has been used for CA based noised filtration [14].

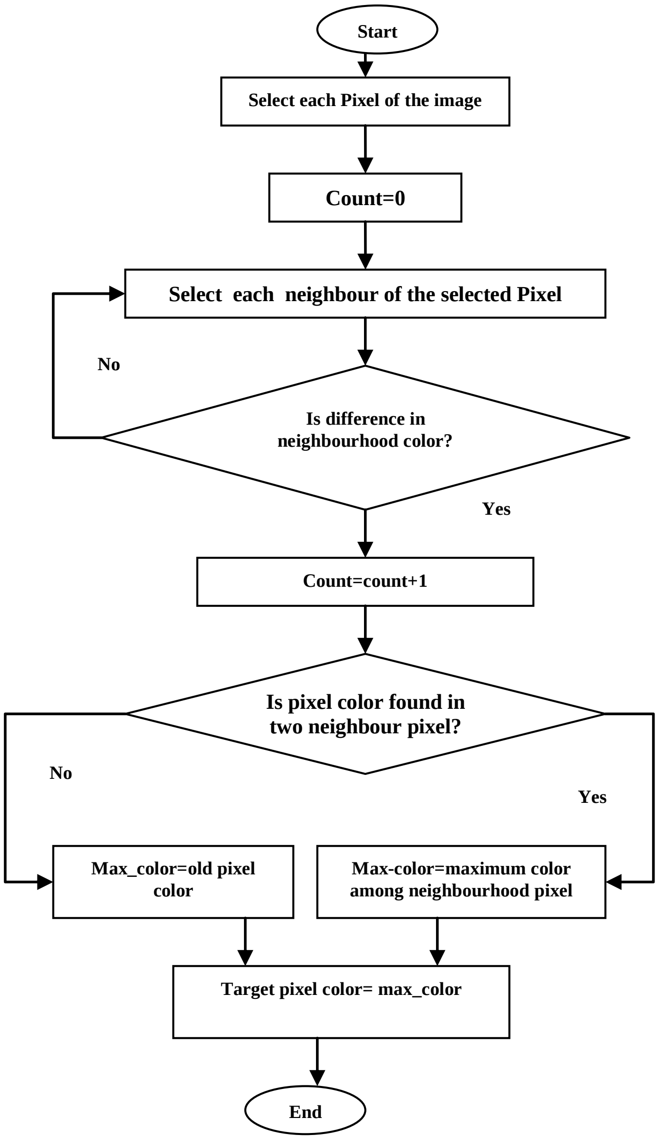


**Figure 4. Proposed Moore Neighborhood CA Noise Filter**

**Algorithm 1: Noise Reduction**
Input: Image with salt and peeper noise
Output: Noise reduced Image
Step 1: Start
Step 2: Initialize Moore neighborhood
Step 3: For all pixel of the Image Compare every pixel color value with surrounding pixels' color value
Step 4: If at-least two matches is found then goto step 7 else goto Step 5
Step 5: Find and store the color value which is maximum in neighborhood pixels
Step 6: Update with new value
Step 7: Goto Step 3
Step 8: Exit

This peeper-salt CA noise filter algorithm has further used to calculate image rank in the flowchart of Figure 5 [15].

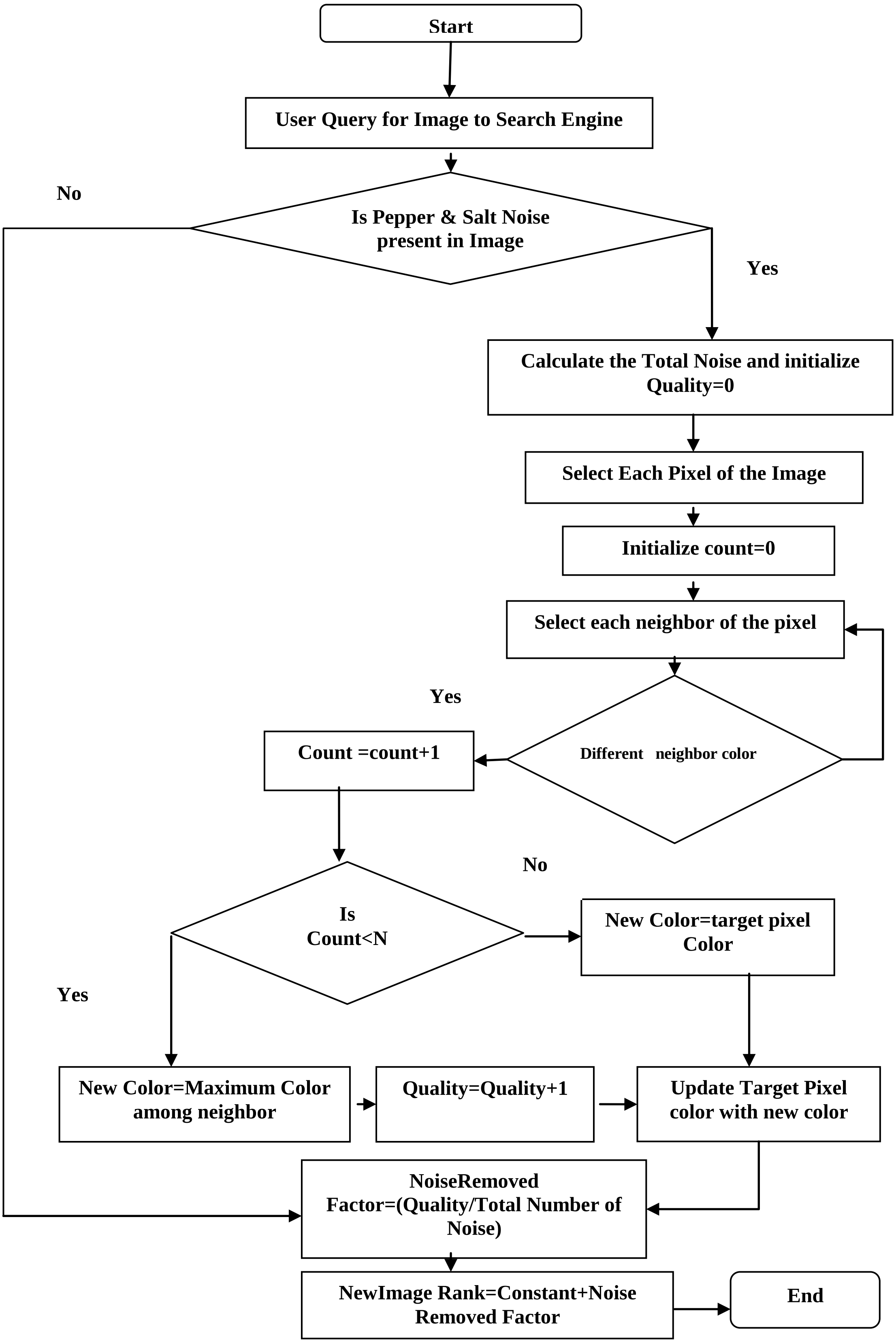


**Figure 5. Flowchart of proposed image-rank calculating system**

Based on the proposed system's flowchart in Figure 5, following Algorithm 2 has been used to calculate the Image-Ranking [15].

***Algorithm 1: Improved Image-Rank***
*Input: User query for Image*
*Output: Improved Image-Ranking*
*Step 1: Start*
*Step 2: If peeper & salt noise is present in required image then follow Step 3 else follow Step 11*
*Step 3: Calculate the total number of Noise N1 in Image and find the Histogram of the Image=* $C_1$
*Step 4: Initialize Moore neighborhood with radius R and noise level N and damping factor(d)=100*
*Step 5: Compare color value of every pixel with surrounding* $((2R+1)^2-1)$ *pixels' color value*
*Step 6: If at-least N matches is found then go to Step 10 else go to Step 7*
*Step 7: Find and store the color value which is maximum in neighborhood pixels*
*Step 8: number of noise removed n=n+1*
*Step 9: Update with new value*
*Step 10: Go to Step 5 for next pixel*
*Step 11: Noise removal factor =* $(N_1-n)/(C1+n)$

*Step 12: New Image-Rank =1+ ┌( Noise Factor *d)┐*
*Step 13: End*

In image-ranking calculation using content based image retrieval the following three basic low level features are used:
i) Color
ii) Shape
iii)Texture
In our approach, we have focused only on the color which is directly based on Color Histogram of an Image. Color Histogram is nothing but the graphical representation of the amount of pixels present in the image of the particular colors. Image-ranking is directly proportional to a factor which is obtained by Noise and Histogram Matching ratio. It implies that the Equation 1 of image-ranking = k*(Noise/Histogram Amount)…(1) where ' k' value depends on Shape and Texture of Image and it is considered as a constant value in our case.
Assume that the amount of histogram $=C_1$ where C1=number of pixel present of a particular color in a image and Noise= $N_1$
Noise Factor $_{prev} = k*(N_1/C_1)$
After a degree of noise reduction, it is found that number of pixel present of a particular color in an image increased to $C_2=C_1+n$ and Noise reduced to $N_2=N_1-n$ where 'n' number of noise has been reduced.
Noise Factor $_{new}=k*(N_2/C_2)$
$=k*((N_1-n)/(C_1+n))$

$$\frac{\text{Noise Factor}_{prev}}{\text{Noise Factor}_{new}} = \frac{(C_1+n)*N_1}{(N_1-n)*C_1} = 1+\frac{(C_1+N_1)*n}{(N_1-n)*C_1} \quad \ldots(2)$$

Now clearly from the Equation 2, the above stated ratio is greater than 1.
So, New Page rank is obviously better than previous

## IV. Experimental observations & Result Analysis

From the experimental results, the following outputs have been observed after several iterations using proposed methodology. Following Figure 6, Figure 7 and Figure 8 shows the different screen shots of experimental procedure.

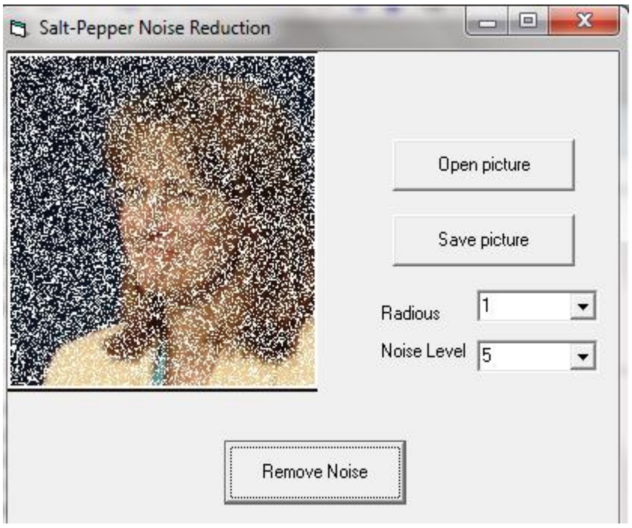


Figure 6: Noisy Input Image

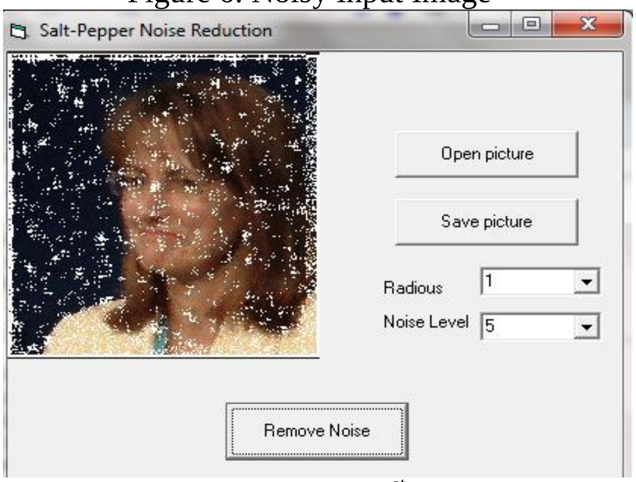


Figure 7: Image after 1st Iteration

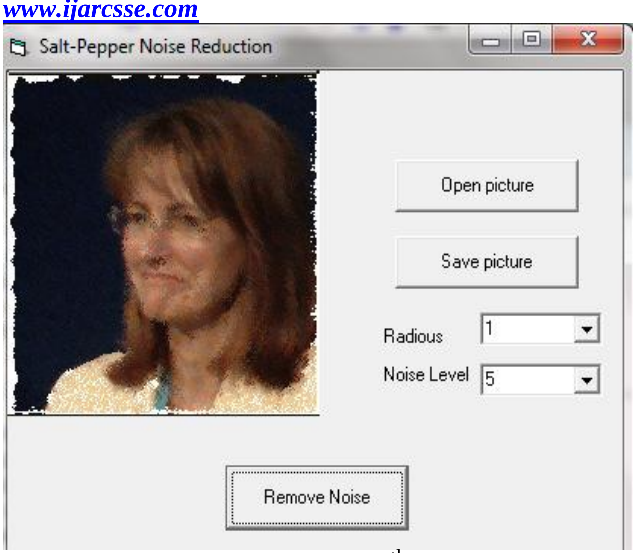


Figure 8: Image after 4th Iteration

The following Figure 9 represents the clarity of the input image, where some noise have been removed after some iterations.

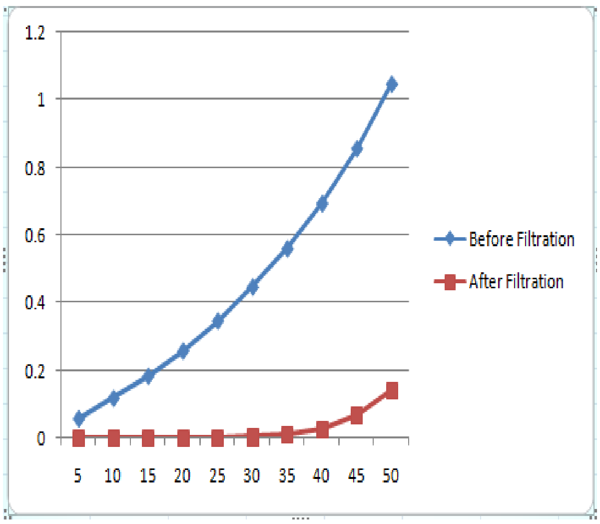


Figure 9: Image clarity improvement after noise removal

The Following Comparison has made with Median Filter with our CA Filter:

**Original Median Filter CA Filter**

**Figure 7: Performance comparison with Median noise filter**

Based on our methodology, Table 2 shows the results of different image-rank of different inputs, observed in our theoretical experiments [15].

Table 2: Image-Ranks

| **Noise Amount ( in %)** | **OLD NOISE FACTOR($N_1$)** | **Old Image Rank** $[1+\lceil N_1*100\rceil]$ | **NEW NOISE FACTOR ($N_2$)** | **New Image Rank** $[1+\lceil N_2*100\rceil]$ |
|---|---|---|---|---|
| 0 | 0 | 1 | 0 | 1 |
| 5 | 0.0588 | 7 | 0 | 1 |
| 10 | 0.1191 | 13 | 0 | 1 |
| 15 | 0.1849 | 20 | 0 | 1 |
| 20 | 0.2592 | 27 | 0.0001 | 2 |
| 25 | 0.3480 | 36 | 0.0004 | 2 |
| 30 | 0.4484 | 46 | 0.0025 | 2 |
| 35 | 0.5619 | 58 | 0.0090 | 2 |
| 40 | 0.6954 | 71 | 0.0262 | 4 |
| 45 | 0.8581 | 87 | 0.0662 | 8 |
| 50 | 1.0489 | 106 | 0.1398 | 15 |

## V. CONCLUSION

From Figure 3, 4 and 5, it is reflected that peeper salt noise can be removed efficiently using CA based noise filter. Figure 6 ensures the smoothness of the image which is reproduced with lesser noise using our CA based noise filter. Figure 7 shows its efficiency over popular median filter. Thus these results and data of Table 1 ensures that the proposed methodology can efficiently be embedded with search engines which will produce better image-rank for noisy image by reducing the noise level.

## REFERENCES

[1] Page_Ranking

http://en.wikipedia.org/wiki/Page_Rank

[2] (2003) , "*Pointing Gesture Recognition based on 3DTracking of Face, Hands and Head Orientation*" ICMI'03, Vancouver,BritishColumbia,Canada.
[3] Image Noise http://en.wikipedia.org/wiki/Image_noise

[4] Wolfram S. (2002)," *A New Kind of Science*"; Champaign, IL, Wolfram Media.

[5] V.R.Vijaykumar, P.T.Vanathi, P.Kanagasabapathy, "*Fast and Efficient Algorithm to Remove Gaussian Noise in Digital Images*"; IAENG International Journal of Computer Science 37:1, IJCS_37_1_09

[6] Chik Ching Yiu, Ip Che Yin, "*Image Ranking Schemes Using Link-Structure Analysis Algorithm*" http://www2002.org/CDROM/poster/114/

[7] Behjat Siddiquie, Rogerio S. Feris, Larry S. Davis, "*Image Ranking and Retrieval based on Multi-Attribute Queries*, http://www. rogerioferis.com/publications/marr_cvpr2011_camReady .pdf

[8] S. A. Chhabria, Ranjana S. Shende, (2011) "*Cellular Automata Image Extraction Algorithm for Gestures Recognition*", IJCST Vol. 2, Iss ue 2.

[9] Paul L. Rosin, (2006) "*Training Cellular Automata for Image Processing", IEEE transactions on image processing",* vol. 15, no. 7.

[10] M. Ajallooeian, A. Borji, B. N. Araabi , M. Nili Ahmadabadi, H. Moradi, (2009) "*Fast Hand Gesture Recognition based on Saliency Maps: An Application to Interactive Robotic Marionette Playing*" The 18th IEEE International Symposium on Robot and Human Interactive Communication Toyama, Japan, Sept. 27-Oct. 2, IEEE.

[11] Adriana Popovici,Dan Popovici, "*Cellular Automata in Image Processing*; http://www-ics.acs.i.kyoto-u.ac.jp/mtns2002/papers/17761_4.pdf

[12] Mr. Chetan A. Burande, Prof. Raju M. Tugnayat, Prof.Dr. Nitin K. Choudhary, "*Advanced Recognition Techniques for Human Computer Interaction.*"

[13] Jigna Bhatt, Niels Da Vitoria Lobo, George Bebis, (2003) "*Automatic Recognition of a Baby Gesture* ", Proceedings of the 15th IEEEInternational Conference on Tools with Artificial Intelligence (ICTAI'03)".

[14] Arnab Mitra, Sourav Samadar, Soumik Sur, Sourav Dey**,** Sayan Chakraborty, (2012) "*Image-Noise-Filtration using 2-D Cellular Automata*", National Conference on Pevasive Computing and Communications (NCPCC),Kolkata.

[15] Sourav Samaddar , Soumik Sur, Arnab Mitra, (2012) "*An intelligent approach to improve Image Ranking based on Image Noise Elimination*", The third All India Academic Meet 2012 by Forum of Scientists, Engineers & Technologists (FOSET), Kolkata.