www.ijarcsse.com

Volume 2, Issue 1, January 2012 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

Sophisticated Image Encryption Using OpenCV

Ashish Pant
Assistant Professor; Computer Sc. & Engg dept.
Dehradun Institute of Technology
Dehradun, India.

Arjun Arora
Assistant Professor; Computer Sc. & Engg dept.
Dehradun Institute of Technology
Dehradun, India.

Suneet Kumar
Assistant Professor; Computer Sc. & Engg dept.
Dehradun Institute of technology
Dehradun, India

Prof. R P Arora
Professor; Computer Sc. & Engg dept.
Dehradun Institute of Technology
Dehradun, India

***Abstract*— With the advancement of Internet the image transferred within the network should be encrypted so that the hackers can not extract the useful information from the image. By using open source computer vision library (OpenCV for short), the data structure of IPL Image (IPL is the main data type of OpenCV which represents image) and its member variable are analyzed and the basic library functions for image handling and processing are used. Library functions are used for loading image, creating a window, saving image, creating an image and to access pixels of image in spatial domain. Arnold transformation also called cat-face transformation is used for first transforming the coordinates of the pixels which is called location scrambling. After that multi dimensional Arnold transformation is used for color scrambling i.e. changing the pixel values for the Red, Green and Blue channels of the image, hence encrypting the image. Inverse Arnold transformation is used for decrypting the image and restoring original image.**

Keywords-: OpenCV; Arnold Transformation; Image Encryption

I. INTRODUCTION

With the rapid development of the computer and communication technology, the means of communication is undergoing profound changes; the traditional means of communication has increasingly become unable to meet the requirements of long-distance, face to face and real-time communication. With the unique advantage image information has got more and more attention. With the development of computer network and communication technology, image transmission can be widely applied to various fields. To ensure the security of image in the network transmission image encryption has become present research focus of domestic and foreign scholars [2-3]. According to encryption method, image encryption can be divided into traditional encryption methods such as DES, RSA, and non-traditional encryption methods such as the use of quantum theory and chaos; according to the encrypted content, it can be divided into direct encryption pixel by pixel and selective encryption[3]. Direct encryption pixel by pixel is to change the original pixel value, which can be applied in image[4]. In this paper, it is different with the others to implement image encryption by OpenCV auxiliary library

II. INTRODUCTION TO OPENCV

.

OpenCV is an open source computer vision library. The library is written in C and C++and runs under Linux, Windows and provides interfaces for Python, Ruby, Matlab and other languages. OpenCV library contains abundant advanced math functions, image processing functions, and computer vision functions that span many areas in vision.

A. *Basic Class*

OpenCV 1.0 includes the following five modules [4]:

1) CxCore: Some basic functions (various data types and basic operations, etc.).
2) CV: Contains image processing and computer vision function(image processing, structure analysis, motion analysis, and object tracking, pattern recognition, and camera calibration).
3) CvAux: Some experimental functions (View Morphing, Three-dimensional Tracking, PCA, HMM).
4) HighGUI: Contains user interface GUI and image/video storage and recall.
5) CvCam: Camera interface (After OpenCV 1.0 version, CvCam will be completely removed.).

B. Environment Configuration

Under windows vista using visual studio 2005 to call the library of OpenCV, my setting steps are as follows:

The first step, download and install OpenCV. Download the appropriate version according to the operating system. For Linux, the source distribution is the file opencv-1.0.0.tar.gz; for windows, you want OpenCV_1.0.exe.

The second step, it is necessary for global settings in Visual Studio 2005:

1) In Visual Studio, choose "Tools->Options";
2) In popup dialog, then choose "Projects and Solutions -> VC++ Directories";
3) In the above dialog, from drop-down list box "Show Directories for:" select "Library files";
4) In Library files listing, add such a path as "C:\Program Files\OpenCV\lib";
5) From 2) dialog's drop-down list box "Show Directories for:" choose "Include Files"; then add the following directory:
"C:\Program Files\OpenCV\cv\include"
"C:\Program Files\OpenCV\cxcore\include"
"C:\Program Files\OpenCV\otherlibs\highgui"
"C:\Program Files\OpenCV\cvaux\include"
"C:\ProgramFiles\OpenCV\otherlibs_graphics\include"
6) Choose "source files" of the drop-down list box "Show Directories for:", then add the following paths to it:
"C:\Program Files\OpenCV\cv\src"
"C:\ProgramFiles\OpenCV\cxcore\src"
"C:\Program Files\OpenCV\cvaux\src"
"C:\Program Files\OpenCV\otherlibs\highgui"
"C:\ProgramFiles\OpenCV\otherlibs_graphics\src"

Doing it like this, the global variable has been set well in Visual Studio 2005. The third step, we can create an project. For example, create a project named OpenCV Video Encryption with"Win32 Application". Do not forget to include the following several header files which should be put behind stdafx.h, otherwise it will go wrong. They are #include <cv.h>, #include<cxcore.h>,
#include<highgui.h> and #include <cvcam.h>.

If lucky, we can compile successfully now. Maybe there are some link errors, so we need to open "Project" -> "Properties" and add the following lib libraries to "Linker" -> "Input" -> "Additional Dependencies", including cxcore.lib, cv.lib, highgui.lib, cvaux.lib and cvcam.lib.

III. HOW TO WOK WITH IMAGE

Analyzing the IPLImage Structure

A M × N image is stored orderly in memory according to the form of the matrix data stored, but the concept of a matrix in OpenCV is somewhat more abstract than the concept we learned in our linear algebra class. In particular, the elements of a matrix need not themselves be simple numbers. For example, a 24-bit color image needs a three-channel two-dimensional matrix to represent red, green, blue bytes. Because the channels are contiguous in a multichannel matrix, the matrix data is stored as: rgbrgbrgb…, gbrgbrgbr…or bgrbgrbgr… and each matrix data means three 8-bit color value. Pixels of image are arranged as shown :

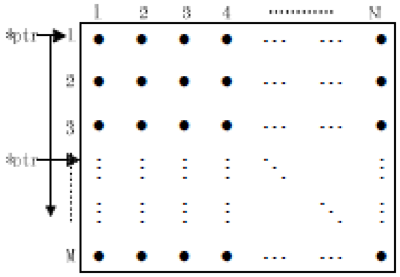

Fig.1 M × N Pixels

Assume integer y and x. They signs row variables and column variables respectively. In that structure, columns or width alone is not enough to move between matrix rows because matrix or image allocation is done to the nearest four-byte boundary. For instance, a matrix of width three bytes would be allocated four bytes with the last one ignored. The widthStep is the length of a row in bytes of a row in the matrix. For this reason ,we must use the widthStep of the matrix to obtain correct offset[4].The following definition of a uchar ptr pointer makes it always point to the first starting position y_{th} row .

*uchar*ptr=(uchar*)(img->imageData+y*img-> widthStep);*

As the x value increased one by one it points to each imageData when pointer ptr has pointed specific line. ptr[3x_i], ptr[3x_i+1] and ptr[3x_i+2] point to the value of three channels of each imageData. The corresponding sequence of the three channels is BGR here. So we would like to change the G values and call the ptr with corresponding subscript directly.

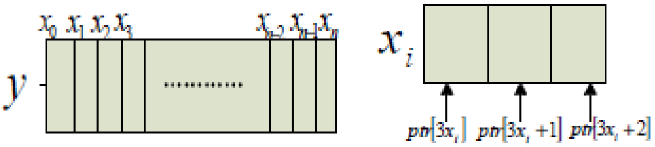

Fig.2 An Image Data

IV. ARNOLD TRANSFORMATION AND APPLICATION

A. Arnold Transformation

Arnold transformation was proposed in ergodic theory. We call it cat-face transformation whose original meaning is cat mapping 1. Suppose that there is a point (x, y) in the unit square. The transformation that changes it into another point (x', y') is something of the form:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 2 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \end{bmatrix} \operatorname{mod}(1) \qquad (1)$$

The transformation is called two-dimensional Arnold transformation (Arnold transformation for short)[1]. So, in fact, the transformation is a position shifting of points. If a square image is expressed as matrix form, F_{xy} denotes the gray value of the pixels whose coordinate is (x, y). Arnold transformation can change the location of image gray values through transforming the coordinate of the pixels. We can use (1) to achieve the image pixels location scrambling [9].

B. Color Scrambling

Qi Dong-Xu[7] extends Arnold Transformation to multi-dimension. The corresponding transformation matrix is as follows:

$$A_N = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & 2 & 2 & \cdots & 2 \\ 1 & 2 & 3 & \cdots & 3 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & 2 & 3 & \cdots & N \end{bmatrix} \qquad (2)$$

For the vector $(x_0, x_1, x_2,\ldots, x_{N-1})^T$, the transformation is formula (3), which gives a moving mode of discrete grid points in n-dimensional space[1].

$$(x_0', x_1', x_2',\ldots, x_{N-1}')^T = A_N (x_0, x_1, x_2,\ldots, x_{N-1})^T \qquad (3)$$

First, take the transposition of the color values as a column vector and they are from three channels all the pixels data of an arbitrary row.

Second, we can get an N by 1 column vector that multiplies matrix A_N on the left and the result is a new N by 1 column vector. The matrix A_N can multiply the result generated by the last operation [5]. It is critical that we transform RGB three channels at the same time.

At last, take modulus of the 256 on each element of N by 1 column vector and assign the results of the modular operation to the corresponding pixel data in turn as the new value of color.

The results of encryption are shown below:

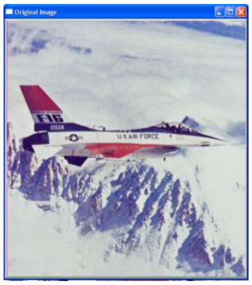

Fig. 3 Original Image

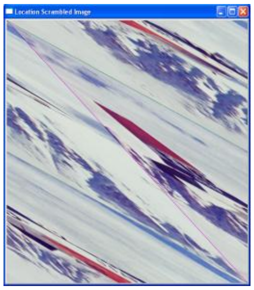

Fig. 4 Location Scrambled Image

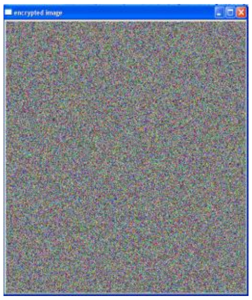

Fig.5 Encrypted Image (Color Scrambling)

V. INVERSE ARNOLD TRANSFORMATION FOR DECRYPTION

The decryption is achieved by applying Inverse Arnold transformation to the encrypted image. The corresponding two dimensional Inverse Arnold transformation matrix is as follows:

$$A_N^{-1} = \begin{bmatrix} 2 & -1 \\ -1 & 1 \end{bmatrix}$$

The three and four dimensional matrices are given as:

$$A_N^{-1} = \begin{bmatrix} 2 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 1 \end{bmatrix} \qquad A_N^{-1} = \begin{bmatrix} 2 & -1 & 0 & 0 \\ -1 & 2 & -1 & 0 \\ 0 & -1 & 2 & -1 \\ 0 & 0 & -1 & 1 \end{bmatrix}$$

Similarly by symmetry N-dimensional Inverse Arnold transformation matrix can be found. The Decryption process is reverse of encryption. For the vector $(x_0', x_1', x_2',\ldots, x_{N-1}')^T$, the inverse transformation is formula (4), which gives a mode of discrete grid points in n-dimensional space and hence the original pixels values are recovered.

$$(x_0, x_1, x_2,\ldots, x_{N-1})^T = A_N^{-1}(x_0', x_1', x_2',\ldots, x_{N-1}')^T \qquad (4)$$

The result of decryption is shown below:

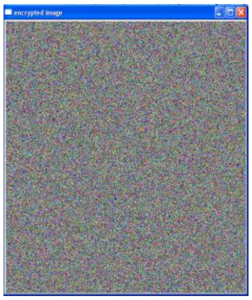

Fig.6 Encrypted Image

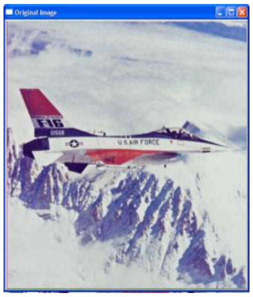

Fig.7 Original Image after Decryption

VI. CONCLUSION

In this paper, IplImage data structure of the OpenCV has been analyzed in detail. In order to make various operations to the image data easily, we utilize the defined pointer to traverse all the image data. Combining with the existing Arnold transformation, the image has been encrypted. We use the inverse Arnold transformation for decryption The library functions of OpenCV make the encryption process simple and feasible, which lay a foundation for trying more updating operations. However, Arnold transformation is a complete and simple method of image encryption and the processing speed will become slow with increasing pixel data. In future, we can do better in improving the speed of encryption and decryption.

REFERENCES

[1] Cnyan Meng, Xinghui Zhang, Video Encryption Based on OpenCv, In: 2nd IEEE International Workshop on Database Technology and Applications ,2010, pp .1-4

[2] ManYoung Rhce. Cryptography and Secure Communication. McGraw-Hill BookCo. , 1994.

[3] Pabien A.P. Petitcolas, Ross J. Anderson and Markus G. Kuhn. Information Hiding-A Survey. Proc. Of IEEE, 1999, 87(7): 1062-1078.

[4] Shiqi Yu, Ruizhen Liu Translate. Learning OpenCV [M]. Beijing: Tsinghua University Press,2009

[5] Ding Wei, Yan Weiqi, Qi Dong-Xu. Digital Image Scrambling Based on Arnold Transformation[J].Computer Aided Design and Computer Graphics Journal,2001,13 (4) :339—341.

[6] Chen Shengyong, LiuSheng etc.. Computer Vision Technical Realization Based on OpenCV [M]. Beijing: Science Press, 2009.

[7] QiDong-Xu, Zou Jiang-Cheng, Han Xiao-You. A new class of scrambling transformation and its application in the image information covering.Science in China(Series E),2000, 43(3): 304-312

[8] V I Arnold, A Avez. Ergodic Problems of Classical Mechanics. Mathematical Physics Monograph Series. New York: W A Ben-jamin, Inc., 1968

[9] RenHonge, ShangZhenwei, ZhangJian. A Digital Image Encryption Algorithm Based on Arnold Transformation [J]. Optical Technology, 2009,35(3):384—390