

Concealment of Targeted Regions in Digital Images on Mobile Devices

Gevorg A. Karapetyan and Hakob G. Sarukhanyan

Institute for Informatics and Automation Problems of NAS RA
e-mail: gevorgka@gmail.com, hakop@ipia.sci.am

Abstract

In this paper we present a framework for robust and effective inpainting of targeted regions in digital images. In framework the modification of frequency selective extrapolation algorithm is used, which provides good quality of concealment. The results of the concealment algorithm of the framework are being compared with the results of the state of art inpainting methods which are included in openCV library available for mobile devices. The framework is implemented for Android OS. For selection of targeted regions in digital images the advance of Bézier curves is used. The effectiveness of the framework has been tested on various digital images.

Keywords: Image inpainting, Regions concealment, Selective extrapolation, Bezeir curve, Mobile devices.

1. Introduction

Mobile computing technologies are growing very rapidly. New mobile devices have multicore processors, which give ability to perform computer vision algorithms with high complexity on mobile devices. This paper describes the implementation of inpainting algorithm on a mobile phone with Android OS.

One of the major uses of mobile devices is taking images via camera of device. Before sharing the images via social networks users usually filter the images for enhancing the image quality as well as editing the image to remove or hide targeted regions. Since the number of shared images of active user is large the application which he or she uses should be user friendly and have high performance. Thus, we have designed an application with user friendly interface and optimized the accurate inpainting algorithm for working fast on a mobile device. For mobile applications several inpainting algorithms [5, 6] are included in openCV (Open Source Computer Vision Library) library [7]. These algorithms can be used in mobile applications via openCV library. OpenCV was built to provide a common infrastructure for computer vision applications and to accelerate the use of machine perception in commercial products. OpenCV library has more than 2500 optimized algorithms, which includes a comprehensive set of both classic and state-of-the-art computer vision and machine learning algorithms. We have explored inpainting methods which are included in openCV lib. These methods are Method by Alexandru Telea [5] and Navier-Stokes [6] based method. Telea inpainting algorithm is based on propagating an image smoothness estimator along the

image gradient. A similar approach is used in [8], where the image smoothness information is estimated by the image Laplacian and is propagated along the isophotes directions, estimated by the image gradient rotated 90 degrees. In Telea algorithm the image smoothness is estimated as a weighted average over the known adjacent areas of the pixel to be inpainted. Into missing regions the information is being propagated via fast marching method (FMM) [9]. The other method in openCV is inpainting using the Navier-Stokes equations. In this approach the isophotes are being propagated with the usage of ideas from classical fluid dynamics [6]. Algorithms provide good results for very small areas and high performance, but have issues with structured and large regions.

In our framework the inpainting is done via a modified frequency selective extrapolation (FSE) [1-4] algorithm, which provides good results in structured and homogeneous regions. The algorithm removes targeted areas from digital image with high accuracy. The areas are being selected with the help of Bezeir curves [4]. The mobile screen sensor handles gesture of user's finger and obtains the targeted area mask. Based on targeted areas and mask is being generated and is passed to inpainting algorithm. The algorithm conceals targeted area with usage elements from adjacent areas. The inpainting algorithm is being implemented on C++ programming. The user interface is developed via Java programming language which uses static lib developed on C++.

The paper is organized as follows: in section 2 targeted areas selection is being described. Then in section 3 the concealment method is being described. In section 4 the experiment results are being shown and compared with inpainting algorithm described in openCV 4.3 [7]. The work concludes in section 5 where a brief summary of proposed framework and future work is described.

2. Selection of Targeted Regions

The binary mask which highlights the unwanted regions of the image is obtained with user interaction. User with finger selects the unwanted region. The selection curves are drawn with the help of Bezeir curves. On screen touch we can obtain the coordinates of current point $P_0(x_0, y_0)$. Then on finger move we measure the distance of the current point from the starting pixel. If it equals to 5, then we store that value and mark it as $P_1(x_1, y_1)$.

$$distance = \sqrt{(x_0 - x_1)^2 + (y_0 - y_1)^2}.$$

Then the user continues the movement of his finger we calculate the distance of ongoing point from P_1 . When the distance is equal to 5px we store that point as P_2 . The set of points P_0, P_1, P_2 are used for building $B(t)$ Bezeir quadratic curve, which is calculated below:

$$B(t) = (1-t)^2 P_0 + 2t(1-t)P_1 + t^2 P_2, \quad t \in [0,1].$$

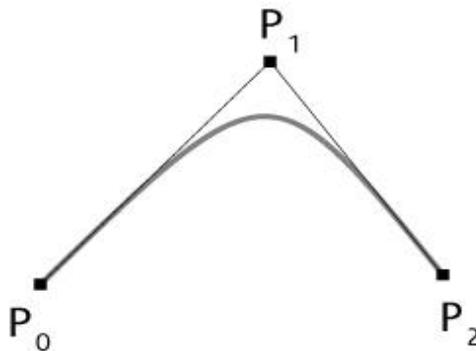


Fig. 1. Bezeir curve built with usage of P_0, P_1, P_2 points

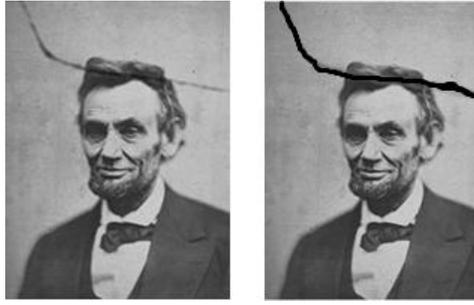


Fig. 2. The result of selection of targeted image areas via Bezeir curves:
a) input image; b) selected targeted region

Then P_2 is set as P_0 and the curve building procedure continues till the user takes the finger up. After selection regions of interest for being concealed we generate binary mask which will be used in concealment algorithm.

3. Concealment Method

Image inpainting is the process of concealing missing data in digital images, via interpolation of the missing pixels using information of the adjacent areas. As an inpainting method we have used a modification of powerful Frequency Selective Extrapolation Algorithm [1-3], where the size of support area is being chosen content aware.

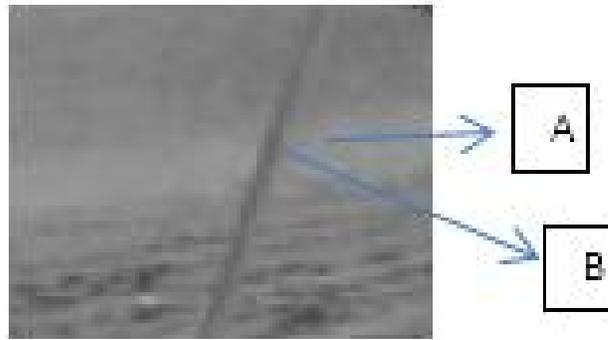


Fig. 3. A - targeted area which should be concealed with help of B supporting area

We choose Γ area, which is composed of corrupted region A and support areas elements from B . The image in Γ we denote $\{f[m,n]\}$, $m = \overline{0,M}$, $n = \overline{0,N}$, where M and N are width and height of area Γ . For estimation of corrupted regions the samples in support area B will be approximated by weighted linear combination of two-dimensional DFT (Discrete Fourier Transform) basis functions, which are defined over area Γ . The approximation will be done with usage of $g[m,n]$ parametric model.

$$g[m,n] = \sum_{(k,l) \in K^v} c_{k,l} \{_{k,l} [m,n],$$

where K^v is the set of basis functions, $c_{k,l}$ is the expansion coefficient, $\{_{k,l}$ 2D DFT basis function which is defined in entire area Γ .

We assume that in any iteration ν the signal is being approximated by $g^{(\nu)}[m, n]$ parametric model, which is calculated as:

$$g^{(\nu)}[m, n] = \sum_{(k,l) \in K^\nu} c_{k,l}^{(\nu)} \xi_{k,l}[m, n],$$

where $c_{k,l}^{(\nu)}$ is the expansion coefficient calculated in iteration ν .

The quality of extrapolation is controlled by calculation of the residual error signal in area Γ , which is calculated by

$$r^{(\nu)}[m, n] = (f[m, n] - g^{(\nu)}[m, n])w[m, n],$$

where $w[m, n]$ is the window function, which is 0 in highlighted area and is 1 in support area, i.e.

$$w[m, n] = \begin{cases} 0, & (m, n) \in A, \\ 1, & (m, n) \in B. \end{cases}$$

For each iteration $E_B^{(\nu)}$ error criterion is being calculated

$$E_B^{(\nu)} = \sum_{(m,n) \in \Gamma} (f[m, n] - g^{(\nu)}[m, n])^2.$$

The iteration stops, when E drops below the pre-defined threshold. After termination of the iterative process the corrupted pixels in area A of $f[m, n]$ are being replaced with the corresponding pixels from $g[m, n]$ parametric model.

The algorithm is working in frequency domain and the algorithm steps are as follows:

1. Initially the $g[m, n]$ parametric model is equal to 0:

$$g^{(0)}[m, n] = 0, \quad \forall (m, n) \in \Gamma.$$

And the residual error is equal to the input signal multiplied with the window function,

$$r^{(0)}[m, n] = f[m, n]w[m, n], \quad \forall (m, n) \in \Gamma.$$

Since the algorithm works in frequency domain, one DFT is applied on $g^0[m, n]$, $f[m, n]$ and $r^0[m, n]$:

$$\begin{aligned} F[m, n] &= DFT\{f[m, n]\}, \\ G^{(\nu=0)}[m, n] &= DFT\{g^{(\nu=0)}[m, n]\}, \\ R^{(\nu=0)}[m, n] &= DFT\{r^{(\nu=0)}[m, n]\}. \end{aligned}$$

2. Calculation of $\Delta E_A^{(\nu+1)}$ energy decrease computation, the difference of error criterion between ν and $(\nu + 1)$ iterations.
3. Selection of (u, ν) basis function indexes value pair, for which $\Delta E_A^{(\nu+1)}$

$$(u, \nu) = \arg \max \Delta E_A^{(\nu+1)}.$$

4. Update of $c_{k,l}^{(\nu+1)}$ expansion coefficient update.
5. Repeat steps 2 – 4 until the $\Delta E_B^{(\nu+1)}$ drops predefined threshold $E_{\min} = 15$ or the number of iteration becomes more than max number of iterations. We used max number of iterations 100.
6. After termination apply Invers Discrete Fourier Transform on parametric model:

$$g[m, n] = IDFT\{G^{(\nu)}[m, n]\}.$$

Then replace the element from $f[m,n]$ which contain specular reflections with corresponding elements from $g[m,n]$: $f[m,n] = g[m,n]$, where $[m,n] \in A$.

4. Implementation on Mobile Device

The proposed inpainting algorithm is implemented on Samsung Galaxy S4 mobile device which has open source Android OS. The mobile device has Octa Core Processor with 1,6 GHz Quad + 1,2 GHz Quad CPU Speed. The user interface is developed on Java and the inpainting algorithm on c++. The algorithm is used as a static library in Java project. We take the input image from the gallery of camera of a mobile device. We have designed an interface for easy selection of targeted regions of the image, which should be concealed with data form adjacent areas. The experiments have been done on different images on structured and textured areas.

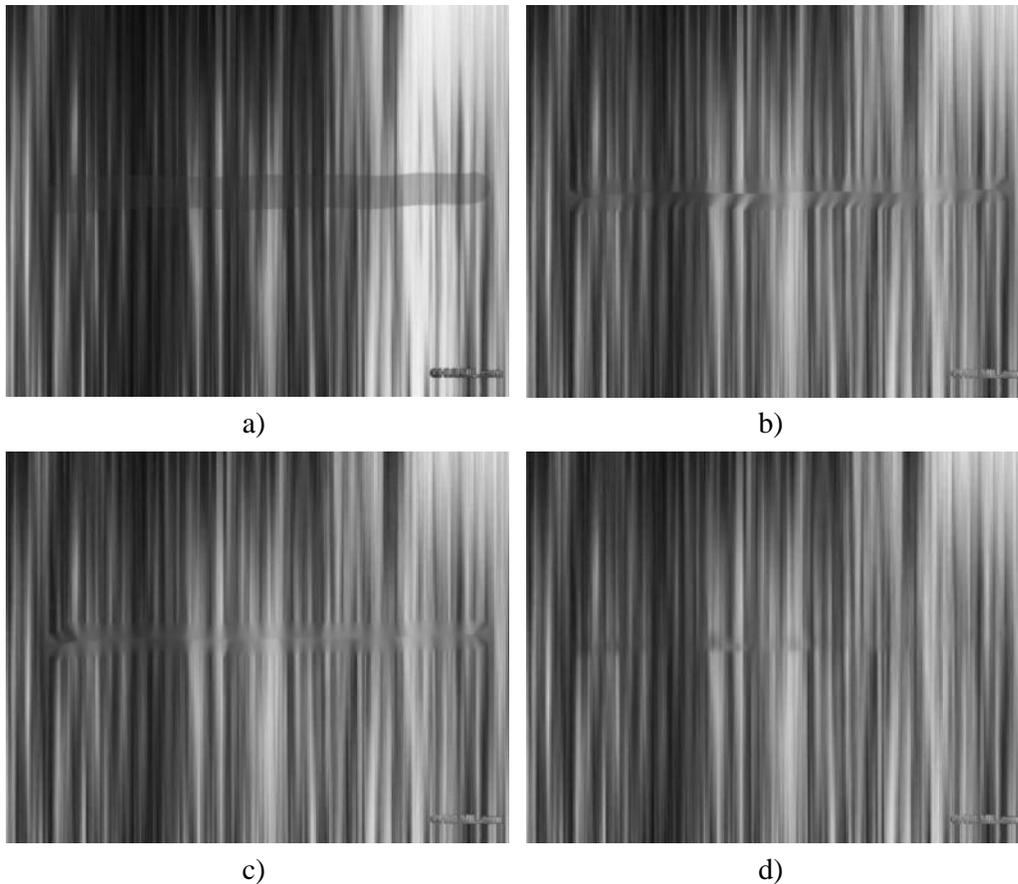


Fig. 5. a) Original image with selected area which should be inpainted; b) Inpainted image with Navier-Stokes method; c) Inpainted image with Telea method; d) Inpainted image with modification of FSE method.

We have also compared inpainting algorithms which are included in openCV library with the algorithm included in our framework. In Navier-Stokes and Telea algorithms the radius of circular neighborhood of each point was set 24. The parameter was chosen by the authors for achieving better PSNR. In modification of FSE are used 16x16 blocks; adaptive supporting area size varies from 3-16 px; fft size 64; max iterations number 100. The results of the comparison are shown in Figure 4 and Table 1.

Table . PSNR and performance time comparison of proposed method with Navier-Stokes and Telea inpainting methods

	Navier-Stokes	Telea	Modification of FSE
PSNR	34.2dB	33.7 dB	38.75 dB
TIME	12704ms	15160ms	26731ms

We can see in Table1 the excellence in quality of the proposed algorithm above Telea and Navier-Stokes methods. But the modification FSE performs slower because of more detailed analyses of support area and iterative propagation.

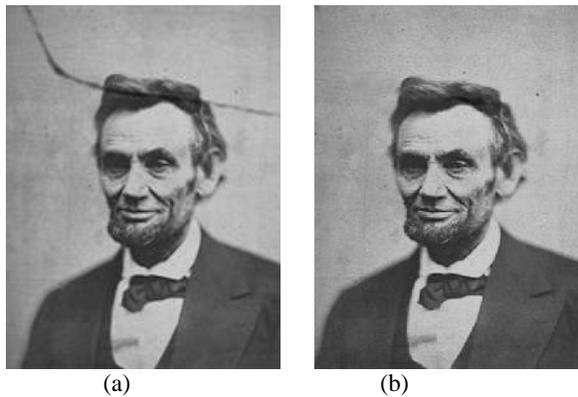


Fig. 4. (a) Original image, (b) Concealed image

Figure 4 shows an example of restoration of an old image via our framework; (a) is the input image which contains a corrupted area and (b) is an output image, where the target areas have been concealed with high accuracy. In Figure 5 an example of removing object form the image is shown; (a) an input image and (b) is an output image where the airplane has been removed.



Fig. 5. a) original image; b) inpainted image with removed object

5. Conclusion

We have implemented a mobile framework for user-interactive selection and concealment of targeted regions in digital images. The framework has been developed for Android OS. The experiments showed the effectiveness and robustness of the proposed framework also provided positive results in comparison with some state of the art inpainting methods which are included in openCV. The modification of FSE provides good results in inpainting not of very large structured and textured areas.

For very large regions the modification of FSE is not effective: we are going to create a hybrid framework which will use an exemplar-based inpainting method for very large regions and modification of FSE for smaller regions. Also we are going to use OpenCL lib for achieving better performance. Moreover, we are going to design an interface for third-party mobile applications which can use our inpainting library.

Acknowledgement

Gevorg Karapetyan would like to thank the State Committee of Science of Armenia for supporting this work by grant 13A-1b30.

References

- [1] A. Kaup, K. Meisinger and T. Aach, “Frequency selective signal extrapolation with applications to error concealment in image communications”, *International Journal of Electronic Communication (AE)*, vol. 59, pp. 147–156, 2005.
- [2] G. Karapetyan, “Modification of FSE method based on coefficients of homogeneity”, *Mathematical Problems of Computer Science*, vol. 35, pp. 109–115, 2011.
- [3] G. Karapetyan and H. Sarukhanyan, “On a modification of the frequency selective extrapolation method“, *Information Models and Analyses*, vol. 2, pp.139–145, 2012.
- [4] T. Farouki, “The Bernstein polynomial basis: A centennial retrospective”, *Computer Aided Geometric Design*, vol. 29, Issue 6, pp. 379 – 419, 2012
- [5] A. Telea, “An image inpainting technique based on the fast marching method”, *Journal of graphics tools* 9.1, pp. 23-34, 2004.
- [6] M. Ebrahimi and M. Holst, “The navier-stokes-voight model for image inpainting”, [Online]. Available: <http://arxiv.org/pdf/0901.4548v3.pdf>
- [7] Open Source Computer Vision Library home page, [Online]. Available: <http://www.opencv.org>
- [8] M. Bertalmio, G. Sapiro, V. Caselles, and C. Ballester. “Image Inpainting”, *Proceedings SIGGRAPH 2000, Computer Graphics Proceedings*, pp. 417–424, 2000.
- [9] J. A. Sethian. “A fast marching level set method for monotonically advancing fronts”, *Proc. Nat. Acad. Sci.*, vol. 9, no.4, pp. 1591—1595, 1996.
- [10] A. Criminisi, P. Perez and K. Toyama, “Object removal by exemplar-based inpainting” *Proceedings of the 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 721--728, 2003.

Submitted 26.08.2013, accepted 15.10.2013.

Թվային պատկերների նպատակային տիրույթների քողարկում շարժական սարքերում

Գ. Կարապետյան և Հ. Սարուխանյան

Ամփոփում

Աշխատանքում ներկայացված է շարժական սարքերում թվային պատկերների նպատակային տիրույթների քողարկման կայուն և արդյունավետ համակարգ: Համակարգում օգտագործված է հաճախականային ընտրովի էքստրապոլյացիայի փոփոխված մեթոդը, որն ապահովում է քողարկման բարձր ճշգրտությունը: Մեր կողմից առաջարկված քողարկման ալգորիթմի արդյունքները համեմատված են քողարկման հայտնի մի շարք ալգորիթմների հետ, որոնք ներառված են շարժական սարքերի համար հասանելի openCV գրադարանում: Մշակված համակարգը աշխատում է Android օպերացիոն համակարգի միջավայրում: Թվային պատկերներում նպատակային տիրույթների ընտրման համար օգտագործվում են Բեզիեի (Bézier) կորերը: Համակարգի արդյունավետությունը ցուցադրված է բազմաթիվ թվային պատկերների միջոցով:

регионов в цифровых изображениях мобильных устройств

openCV,

Android.