参赛队员姓名：戚涵，焦鹏屹

中学：天津市南开中学

省份：天津市

国家/地区：中国

指导教师姓名：程明明

论文题目：基于删除无效内容以及人脸检测的图像缩放研究

# The study of image resizing based on deleting unobtrusive seam and face detection

By: Qi Han, Jiao Pengyi

Instructor: Cheng Mingming

Tianjin Nankai high school, Tianjin, China

Sep. 10th, 2017

# Contents

# Abstract

Nowadays, pictures have more and more applications in various fields. Because different situations may have different commands of the ratios of pictures, we present a program which can resize the pictures according to their contents. Compared with the traditional method which only cares about the geometric constraints, our program can reserve the important contents of the pictures, which means that the visual effect of the pictures will not be damaged after the resizing.

In order to change the ratio of pictures while avoiding distorting their main contents, we use a method called seam carving. Based on the energy function used for finding useless pixels and dynamic programming used for finding some continuous seams which will be deleted later, our program can finally change the image to a new ratio perfectly. Due to some deficiencies of the basic program, we also make some improvements. The first one is the improvement of the algorithm: we add face detection to the basic program. Not only does this improvement can solve many problems of the basic program automatically, but also it is our own innovation of the study of content-aware resizing. The second one is the addition of customization: we allow users to select the specific contents that they want to reserve, which can meet the different needs of users. It's also our own innovation which will be very useful in practical situations.

Our program can have many practical applications. We have some ideas about how to put this algorithm into real use, which is one of our practical innovations of this topic. In terms of different ratios of photo printing, graphic designing and screen displaying, this program will have various uses, which will make a groundbreaking progress in practice.


**Keywords:** content-aware, seam carving, face detection, customization, energy function, dynamic programming

# 1 Introduction

As the saying goes, a map is worth a thousand words. With the progress of science and technology, more and more people are willing to use electronic equipment to record daily life. Usually, our photos are saved in those equipment, such as telephones, iPad, cameras and so on. The pictures shown by the electronical screens are perfect but one day when we want to print out these photos, we will encounter a lot of problems. Because the electronical devices screens' ratios are different and even some photos are vertical screen shot and some photos are horizontal screen shot, it's hard to print these pictures in the same scale unless we accept the pictures with seriously deformed contents.

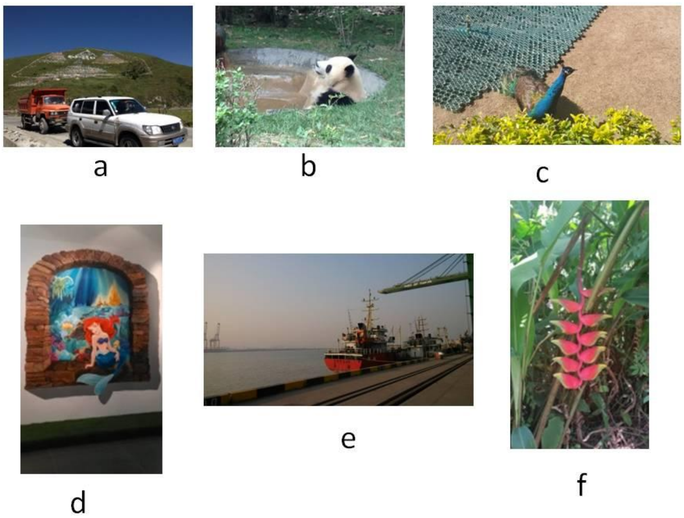


**Fig. 1.1 Photos of different sizes taken by various devices**

( **a & b**, photos that are shot by the iPhone 4, the aspect ratio of 3: 2. **c & f,** photos that are Samsung shot out, the aspect ratio of 16: 9, and f is the vertical screen shooting. **d & e,** photos that are shot by Xiaomi, the aspect ratio is 15: 9, and d is the vertical screen shooting.)

**Fig. 1.2 Photos that are resized only according to their geometric constraints**

(The above pictures are scaled with the conventional method that we stretch the photos wider or narrower directly. We can find the picture content is anamorphic, affecting the visual effects.)

We can further discover that the nature of this problem is how to change the scale of the picture without affecting the content of the picture. In fact, this problem is involved in many fields. Besides this problem in our daily life, we found that the distortions of pictures also bring a lot of troubles in many fields, ranging from webpage making to photography. With the popularity of technology, more and more people will browse the websites with many different devices, such as telephones, iPad, and computers. What's more "terrible" is that all these devices have different sizes of screen. So, the people who design webpage cannot guarantee that all the pictures on the webs will have good-looking on these devices if he or she only uses the ordinary method of image resizing. To the people who are interested in photography, he or she will also face with the problem when the 3*5 picture is converted into 6*6 pictures. To avoid distorting the picture, a lot of people always turn to Photoshop for help. Someone cuts the useless part of the photo by themselves. But this method seems slow and limited, when the useless part is scattering and irregular. Someone, who is familiar with Photoshop, would like use Clone Stamp Tool to change the size. This way, however, will cost at least fifteen minutes to deal with a very ordinary picture, which is in a low efficiency.

But our program is based on Content-Aware Resizing, which can not only reach the same goals stating above automatically, but also it can finish a lot of more complex goals, such as reserving a specific area of a picture, recognizing people in the photos to cut the picture more precisely, and so on. The main algorithm in out project is Energy Function, Dynamic Programming and Face Detection. Tested by about a great number of pictures of various contents, our project can deal with the pictures successfully. For a few pictures which are filled with too many colors and subjects, our project also provides selecting function, which can meet more requirements broached by users.

# 2 Model & Theory

## 2.1 Energy function

When resizing the images, the most important of all is to keep the completeness of the information expressed. Formally, we can consider parts varied with great ranges include more information. Remove unobtrusive pixels that are mixed with their surroundings will not make a difference. To find out the pixels to be removed, we introduce *energy function*.

*For pixel P (x, y), its energy function is* $E(P) = \left|\frac{\partial}{\partial x}P\right| + \left|\frac{\partial}{\partial y}P\right|$.

| | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 136 | 212 | 255 | 255 | 255 | 255 | 255 | 255 | 80 | 152 | 188 | 224 | 208 | 220 | 8 | 104 | 160 | 240 | 232 | 152 |
| 255 | 76 | 255 | 100 | 168 | 66 | 40 | 136 | 6 | 136 | 26 | 255 | 255 | 255 | 238 | 68 | 120 | 255 | 242 | 255 |
| 255 | 70 | 255 | 255 | 255 | 176 | 34 | 104 | 140 | 106 | 255 | 66 | 150 | 74 | 255 | 204 | 164 | 200 | 216 | 228 |
| 255 | 162 | 255 | 255 | 210 | 160 | 170 | 22 | 168 | 255 | 34 | 255 | 255 | 255 | 255 | 242 | 255 | 102 | 140 | 136 |
| 255 | 255 | 255 | 255 | 255 | 255 | 2 | 190 | 255 | 20 | 90 | 204 | 120 | 40 | 255 | 80 | 212 | 255 | 158 | 116 |
| 156 | 26 | 255 | 255 | 255 | 255 | 110 | 204 | 255 | 255 | 238 | 255 | 255 | 255 | 76 | 255 | 255 | 40 | 182 | 12 |
| 255 | 255 | 255 | 255 | 20 | 255 | 255 | 255 | 255 | 255 | 236 | 168 | 255 | 255 | 244 | 134 | 238 | 178 | 142 | 180 |
| 255 | 58 | 74 | 255 | 255 | 186 | 255 | 255 | 255 | 255 | 255 | 34 | 38 | 22 | 68 | 20 | 136 | 54 | 106 | 52 |
| 248 | 102 | 242 | 255 | 70 | 255 | 254 | 255 | 255 | 255 | 88 | 250 | 88 | 30 | 110 | 22 | 44 | 62 | 28 | 124 |
| 40 | 96 | 158 | 60 | 36 | 4 | 152 | 255 | 255 | 255 | 255 | 44 | 104 | 48 | 6 | 100 | 28 | 20 | 38 | 48 |
| 16 | 118 | 36 | 88 | 38 | 60 | 255 | 255 | 194 | 150 | 255 | 240 | 130 | 255 | 70 | 52 | 32 | 38 | 40 | 28 |
| 80 | 14 | 86 | 40 | 74 | 118 | 255 | 255 | 255 | 32 | 140 | 255 | 64 | 38 | 106 | 138 | 84 | 160 | 24 | 36 |
| 56 | 152 | 24 | 212 | 36 | 84 | 136 | 140 | 255 | 255 | 236 | 255 | 255 | 255 | 252 | 152 | 68 | 64 | 112 | 96 |

**Fig. 2.1 A demonstration of the energy power of each pixel calculated by our program**

We can delete same number of pixels from each row to narrow the width of image while keeping the rectangular shape. We should minimize the sum of energy function of pixels removed, thus maximize the amount of information of the remaining parts. But only to do so will destroy the continuity of picture. And only remove pixels of one column whose sum is the smallest at a time will also lead to breakage. So we remove pixels by *seam*, and this is *seam carving*. Define a vertical seam in a n×m image to be:

$$s^x = \{(x(i), i)\}_{i=1}^{n} \quad (s.t. \forall i, |x(i) - x(i-1)| \le 1)$$

And x is a mapping $x: [1,2, \dots, n] \rightarrow [1,2, \dots, m]$.

**Fig. 2.2 Vertical seam and horizontal seam**

## 2.2 Dynamic programming

To explain the core algorithm of this program-dynamic programming-more clearly, we would like to introduce an example. This example is a 5*4 picture as showed below(obviously it's not a real situation, just for explanation).

| | | | | |
|---|---|---|---|---|
| 199 | 5 | 89 | 77 | 60 |
| 76 | 45 | 3 | 80 | 30 |
| 233 | 142 | 10 | 100 | 70 |
| 165 | 1 | 197 | 46 | 255 |

**Fig. 2.3 The sample of a 5*4 picture**

(This form shows the energy power of each pixel.)

To describe the procedure more clearly and conveniently, we firstly define a term *the best sum*. If we want to calculate *the best sum* from pixel[i1][j1] to pixel[i2][j2], we should find one pixel in each row(from row i1+1 to i2-1) and these pixels must be connected; and then we sum up all the energy power of the pixels we choose and the energy power of pixel[i1][j1] & pixel[i2][j2]. If the sum is lowest among all kinds of choices, this sum is *the best sum* from pixel[i1][j1] to pixel[i2][j2].

Then we define the term, the *optimum energy consumption*. If we want to calculate *optimum energy consumption* of pixel[x][y], we should find a pixel in row one to make sure that *the best sum* from this pixel to pixel[x][y] is lowest than any other pixel in row one. And *the best sum* from this pixel to pixel[x][y] is the *optimum energy consumption* of pixel[x][y]. We call the path, through which we can get the *optimum energy consumption*, *the shortest path*.

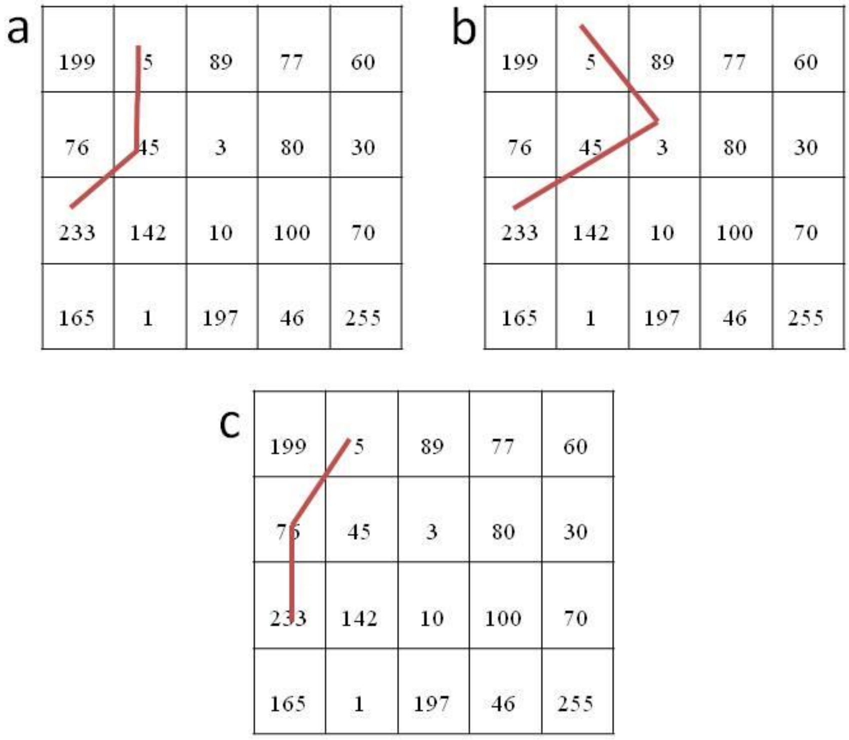


**Fig. 2.4 The correct and wrong samples of the shortest path**

(**a,** the picture shows the correct shortest path of pixel[3][1].**b**, why the red line is not the shortest path is that pixel[2][3] and pixel [3][1] are not connected. **c**, why the red line is not the shortest path is that this path doesn't produce the optimum energy consumption (5+45+233<5+76+233))

In this program, our aim is to find the shortest path of all the pixels in the last row and delete the path which produces the lowest optimum energy consumption among these paths.

We define 3 two-dimension arrays: M[i][j], e[i][j] and path[i][j]. e[i][j] is the energy power of each pixel and M[i][j] is the optimum energy consumption of pixel[i][j]. path[i][j] records the column of pixel in the previous row from which we reach the pixel[i][j] to follow the shortest path.

### 2.2.1 Initialization

For the first row, the optimum energy consumption for each pixel if the pixel's own energy power. So M[1][j] is e[1][j] and path[i][j] is j.

### 2.2.2 Procedure

Starting from the second row, we should calculate the optimum energy consumption for each pixel in this row. Take pixel[2][5] for example. Because the way we choose should be connected, the pixels in the previous row which can reach pixel [2][5] can only be pixel[1][4], pixel[1][5] and pixel[1][6]. So the optimum energy consumption of pixel[2][5] can only be M[1][4]+e[2][5] or M[1][5]+e[2][5] or M[1][6]+e[2][5]. To get the optimum energy consumption, we should just compare that three numbers and let M[2][5] to be the lowest among them. We can write a general equation for all the pixels in this row. $M(i,j) = e(i,j) + min\big(M(i-1,j-$ $1, M i-1, j, M i-1, j+1$

For the other rows, we can follow the equation above.

After calculating, we should also record the path.

### 2.2.3 Find the seam to delete

When we get all the optimum energy consumptions of all the pixels in the last row, we will find the lowest among them and the way which produces that lowest number is the seam we should delete.

# 3 Evaluation of the basic program

## 3.1 Successful samples of the basic program

Our basic program works very well for some natural scenery photographs because these photos generally have a relatively single background (such as blue sky, grass).


**Fig. 3.1 Successful sample I**

(**a,** a 1200*800 picture, which is the original picture waiting for rescaling. **b,** an 800*800 picture, which is the result image of our program. **c,** an 800*800 picture, which is scaled directly. We can find the obvious differences between picture b & picture c. For example, the three people of picture c are squeezed, but in picture b, all the people almost remain their original appearances.)

To think about how our program works when dealing with the pictures, we can find that, for example, pavement, grass, groves, etc. have a smaller energy value, and people, bicycles and wildflowers in the background have a higher energy value. The pixels with smaller energy values are deleted, and the pixels with higher energy values are preserved, so that the desired effect can be obtained.

Below we will list some of the outcome of our program and compare them with the results obtained by direct scaling. We turn all the pictures into square shapes, which are extreme situations of scaling.

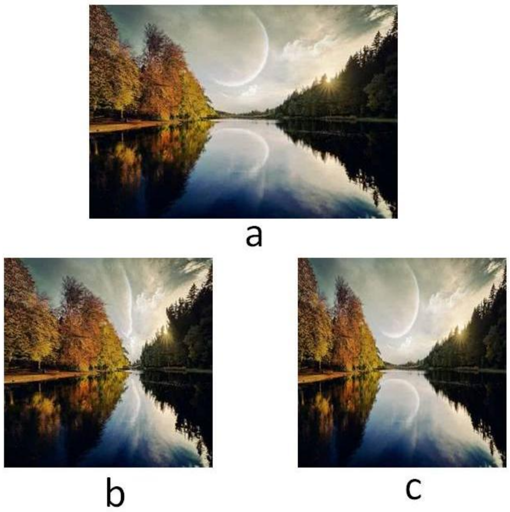


**Fig. 3.2 Successful sample II**

(**a,** the original rectangular picture. **b,** the picture produced by our program. **c**, the picture scaled directly. In our program, we assume that the white sky and blue lake are unimportant information, so they are given very low energy points. So, we will get picture b, in which some parts of the sky and lake are sacrificed but the beautiful trees, as the main objects of this picture, are remained)

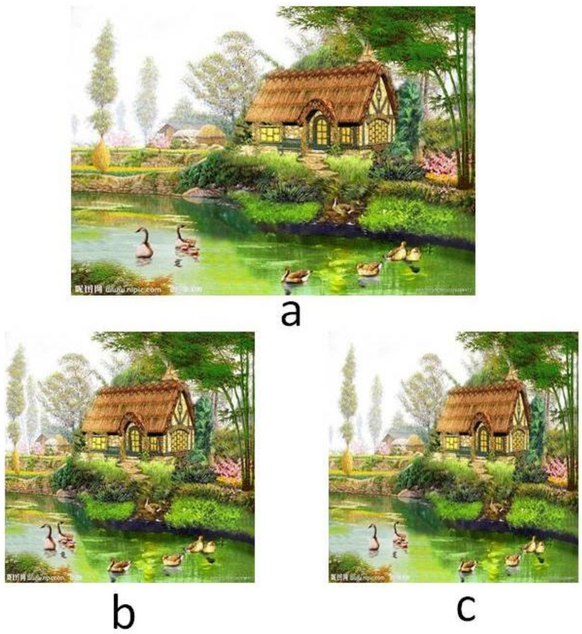


**Fig. 3.3 Successful sample III**

(**a,** the original rectangular picture. **b,** the picture produced by our program. **c**, the picture scaled directly. It's obviously that the house of picture c is malformed, but our program automatedly reserves the whole house but cuts some less pivotal part of this picture)

## 3.2 Problems of the basic program

But for the photos with rich contents (such as those containing both people, which we want to reserve, and very complex background), the basic program is limited. The algorithm of the basic program only calculates each pixel's energy point and delete those seams with low energy power. But for the pictures mentioned above, both the background and main objects have very high energy points, for every part of the picture has very intensified color changing. As a result, the program may be confused when deleting seams and probably will delete the information we want to reserve, resulting in the main deformation of the image. The below pictures are an illustration of this situation.

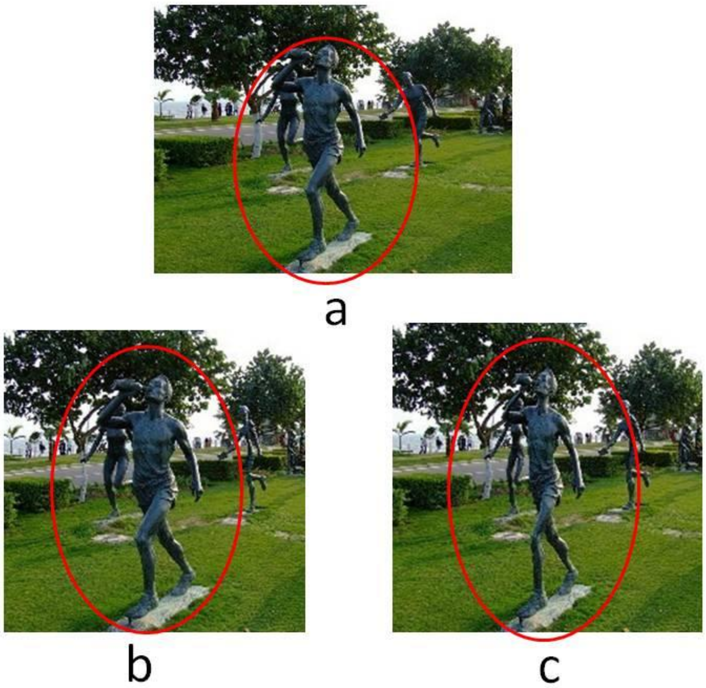


**Fig. 3.4 Failed sample**

(**a,** the original picture. **b,** the picture that we want to get, in which the main sculpture is well reserved. **c,** the outcome of our program in practice)

## 3.3 Improvement 1: face detection


**Fig. 3.5 A picture of face detection**

(The football players' faces in this picture have been detected by out program. The pixels in the blue regions will be assigned to very high energy power so that the faces will not be delete)

We have further improved the procedures for these problems. For human, faces are probably the most significant information in the picture. So, the ability to recognize faces is very necessary. Our program carries face detection algorithm, and give the pixels of the face extra energy to keep them complete. We find that this method is very effective, because the biggest challenge faced by the previous program is the wrong deletion of human faces (or other objects' faces). As is known that the whole face is almost in the same color, such as flesh color, which causes that the whole face will get relative low energy points. What's more, the contours of faces are so pivotal that if the faces are partly deleted, the deformations will be obvious, which is different from deleting part of the skies or grasslands. So, adding face detection algorithm to our basic program can be a good solution to the previous problems. Moreover, the users needn't to have extra troubles when dealing with the pictures. Just let the computer do all the work.

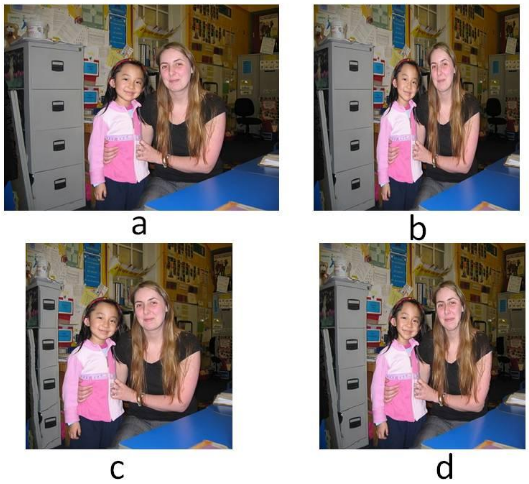


**Fig. 3.6 A successful sample of the improved program**

(**a,** the original picture. **b,** the rescaling picture directly. **c,** the outcome of our improved program, based on face detection. d, the outcome of our basic program, which doesn't work well in this kind of situation)

## 3.4 Improvement 2: users' inputs to guide the program

Different users have different requirement, so solving countless conditions in practical use by the same program is obviously unreasonable. We allow users to select some important parts manually, to make a man-machine interaction. The below picture is a brief show of how this function works.


**Fig. 3.7 Exhibition of the operating window**

# 4 Applications

## 4.1 Photo Printing

Now, more and more people are equipped with a printer at home, in the supermarket is also very easy to buy like paper, but when some photos taken with mobile phone print out, you will find the phone screen length and width ratio and our traditional Like paper, the aspect ratio varies greatly. This is because the mobile phone developers now for the appearance of mobile phone appearance, the screen size has lost the standardization, has not meet our traditional photos a few inches limit. But this is the case, the print out of the photo or leave a blank space, or photos will be due to changes in the aspect ratio of the deformation, the problem has become a lot of family print photos of a major distress. But our program can help users to achieve a good print.

**Sheet. 4.1 Mobile Phones**

| Model | Resolution | Approximate Ratio |
|---|---|---|
| iPhone 4 Series | 960*640 | 3:2 |
| iPhone 5 Series | 1136*640 | 7:4 |
| iPhone 6 & 6S | 1334*750 | 3:2 |
| iPhone 6+ & 6S+ | 1920*1080 | 9:5 |
| Others | 1280*720 or 2560 * 1440 or … | 9:5 |

**Sheet.4.2 Standard Size of Printing Paper (In China)**

| | Size (cm) | Approximate Ratio |
|---|---|---|
| 5 cun | 8.9*12.7 | 5:7 |
| 6 cun | 10.2*15.2 | 5:7 |
| 7 cun | 12.7*17.8 | 5:7 |

Nowadays, there is a kind of machines which are so welcomed in many shopping malls-the free printers. But the users will be somehow disappointed that the photos printed out will always have white margins (like the picture showed below). That's because the different size of photos in our phones. If our program can be applied to this kind of machines, we are sure that these machines will be more popular.

**Fig. 4.1  Photo printer and its products**

Moreover, many people now want to be their photos as WeChat, QQ picture, or print them out, installed in a small pendant as a decoration. In these cases, people usually want to convert the photo into a square, if you take a direct drawing method, will affect the picture content of the aesthetics, so our program will be based on image content to cut, to meet people's ideal for photo scaling requirements.

## 4.2 Picture resizing on different devices

With more and more kinds of technology devised, we can use various devices to take pictures or recording videos. But due to the different sizes of the devices' screens, it's hard to make full screen without distorting the image content when displaying the photos or video on other devices. For example, in our daily life, many families would like to buy wide-screen TVs. But they will find that many objects show by their TVs seem to be "fatter", which damages the visual effect and even go against the basic aim of wide-screen TV-to have better visual effect. If our program can be applied to these situations, these problems will be solved.

## 4.3 Website making and the similar fields.

Having made surveys among those whose jobs are associated with pictures, such as the website makers or even some people who want to add pictures to their papers, essays or presentation, we find that many people are annoyed by a problem: usually the sizes of the original pictures doesn’t meet their need. For example, to a website maker, he will get pictures as the resources of the website from various sources, such as the Internet, photographer’s work and so on. But usually the sizes of these pictures doesn’t fit his design of the website’s composition (for instance, he reserves a square space for the picture but actually he gets a rectangular one). He has to resize the pictures with the conventional method only to get a picture with malformation, which has a bad effect on the visual effect of the website. To a broader context, many people have met the problem of the deformity of picture content due to the resizing with traditional method. 7 of 10 people, according to our survey, said that they always add pictures to their papers or presentations and they are looking forward to an app which can change the size of pictures optionally while reserving the original visual effect. If our program can be made as an app, it will solve many problems and have a broad use.

# 5 Conclusion

We present a method combining seam carving and face detection (or the uses' interactive operation) in order to resize the picture according to its content. The energy function and dynamic programming provide an approach of finding the seam, which keeps the completeness the pictures and meet the geometric constraints. The face detection (or the uses' interactive operation) provides solutions to more specific problems of the basic program, which makes this program more intelligent. Our appealing part is the improvement of the basic program, which solve many problems of the basic program, which are found by ourselves through many times of experiments. Our program can also have many real applications. We have some ideas about how to put this algorithm into real use, which is one of our practical innovations of this topic. In terms of photo printing, picture displaying and website making, this program will have a broad use, which will make a groundbreaking progress in the practices.

# References

1. Avidan S, Shamir A. Seam Carving for Content-Aware Image Resizing[J]. ACM Transactions on Graphics, 2007, 26 (3) :10

2. Agarwala A, Dontcheva M, Agrawala M, Drucker S, Colburn A, Curless B, Salesin D, Cohen M. Interactive digital photo montage[J]. ACM Transactions on Graphics, 2004,23(3): 294–302

3. Shen J, Wang D, Li X. Depth-aware image seam carving[J]. Transactions on cybernetics, 2013,43(5):1453-1461

4. Zhao D, Wang B. Optimized content-aware image resizing with merging and improved importance diffusion[J]. Journal of harbin institute of technology(new series), 2015,22(2):67-73

5.Boykov Y, Jolly M. Interactive graph cuts for optimal boundary & region segmentation of objects inn-dim-ages. International Conference on Computer, 2001,ICCV(1):105–112.

6. Suh B, Ling H, Bederson BB, Jacobs, DW. Automatic thumb nail cropping and its effectiveness. InUIST'03: Proceeding soft the16th annual ACM symposium on User interface software and technology, ACMPress, 2003, NewYork, NY, USA, 95–104

7. Liu H, Xie X, Ma W, Zhang H. Automatic browsing of large pictures on mobile devices. Proceeding soft the eleventh ACM international conference on Multimedia, 2003:148–155

8. Belongie S, Malik J, Puzicha J. Shape matching and object recognition using shape contexts[J]. IEEE Transactions on pattern analysis and machine intelligence, 2002(24):509-522

9. Zhang GX, Cheng MM, Hu SM, Martin RR. A shape-preserving approach to image resizing[J]. Computer Graphics Forum (Pacific Graphics). 2009,28(7):1897-1906

## Acknowledgments

Jiao Pengyi mainly writes about Introduction and Model & Theory Parts; and Qi Han mainly writes about Evaluations of the basic program and Applications. Qi Han provides the source code of this program. We are both responsible for the revision of this paper.

This research is directed by our instructor, Prof. Cheng Mingming, who is the prof. of Nankai University majoring in computer science and image recognition. We have learned the basic method of scientific research and many up to date knowledge about image recognition from him. We would also like to thank the project of “Talent program” carried out by CAST. We are fortunately involved in the project and try to learn how to do research in computer field.

## Introduction of instructor: Prof. Cheng Mingming

程明明，南开大学的计算机与控制工程学院教授，博士生导师，中科协青年人才托举工程、天津市青年千人、南开大学百名青年学科带头人计划入选者。2012 年于清华大学获得博士学位，之后在英国牛津进行计算机视觉研究，并于2014 年加入南开大学。其研究方向主要包括：计算机图形学、计算机视觉、图像处理等。已在 IEEE PAMI 等 CCF-A 类国际会议及期刊发表论文 20 余篇。相关研究成果受到国内外同行的广泛认可，论文他引 5000 余次，最高单篇他引1700 余次。其研究工作曾被英国《BBC》，德国《明镜周刊》，美国《赫芬顿邮报》等权威国际媒体撰文报道。

# Innovation Statement

This paper submitted is the research results under the guidance of instructor. As we know, the paper does not contain research results that have been published or written by other people or teams, except as specifically noted and acknowledged in the text. If there is any dishonesty, we are willing to bear all the relevant responsibilities.

**Team members**: Qi Han, Jiao Pengyi

# 学术诚信声明

本参赛团队声明所提交的论文是在指导老师指导下进行的研究工作和取得的研究成果。尽本团队所知，除了文中特别加以标注和致谢中所罗列的内容以外，论文中不包含其他人已经发表或撰写过的研究成果。若有不实之处，本人愿意承担一切相关责任。

参赛队员： 戚涵 焦鹏屹

指导老师： 程明明

2017 年 9 月 10 日