Fault Tolerance Issue in a New Method for Depth Detection

MAHDI MIRZABAKI
Computer Engineering Department
Azad University of Tabriz, Faculty of Engineering
Azad University Camp of Tabriz
IRAN

Abstract: - There are some different methods used for depth perception. In this paper, the Fault Tolerance issue of a new method for depth perception is considered. This method is based on interpolation. In order to find the parameters of the interpolation function, a set of lines with predefined distance from camera is used, and then the distance of each line from the bottom edge of the picture (as the origin line) is calculated. As it is known, Fault Tolerance is an important property of each method. So in the next part of this paper, this problem has been discussed. The results of implementation of this method show higher accuracy and less computation complexity and more fault tolerance with respect to the other methods. Moreover, two famous interpolation functions namely, Lagrange and Divided Difference are compared in terms of their computational complexity and accuracy in depth detection by using a single camera.

Key-Words: - Depth Detection, Digital Camera, Measurement, Accuracy, Performance, Interpolation, Fault Tolerance

1 Introduction

Depth finding by using camera and image processing, have variant applications, including industry, robotic and vehicles navigation and controlling. This issue has been examined from different viewpoints, and a number of researches have conducted some valuable studies in this field. All of the introduced methods can be categorized into six main classes.

The first class includes all methods that are based on using two cameras. These methods origin from the earliest researches in this field that employ the characteristics of human eye functions. The Main difficulty of these methods is the need to have mechanical moving and the adjustment of the cameras in order to provide proper focusing on the object. Another drawback is the need of the two cameras, which will bring more cost and the system will fail if one of them fails.

The second class emphasize on using a single camera [7]. In these methods, the base of the measurement is the amount of the image resizing in proportion to the camera movement. These methods need to know the main size -of the object subjected to distance measurement and the camera's parameters such as the focal length of its lens.

The methods in the third class are used for measuring the distance of the moving targets [1]. In these methods, a camera is mounted on a fixed station. Then the moving object(s) is (are) indicated, based on the four scenarios: maximum velocity, small velocity changes, coherent motion, and continuous motion. Finally, the distance of the specified target is calculated. The major problem in these methods is the large amount of the necessary calculations.

The fourth class includes the methods which use a sequence of images captured with a single camera for depth perception based on the geometrical model of the object and the camera [8]. In these methods, the results will be approximated. In addition, using these methods for the near field (for the objects near to the camera) is impossible.

The fifth class of algorithms prefers depth finding by using blurred edges in the image [5]. In these cases, the basic framework is as follows: The observed image of an object is modeled as a result of convolving the focused image of the object with a point spread function. This point-spread function depends both on the camera parameters and the distance of the object from the camera. The point-spread function is considered to be rotationally symmetric (isotropic). The line-spread function corresponding to this point spread function is computed from a blurred step-edge. The measure of the spread of the line-spread function is estimated from its second central moment. This spread is shown to be related linearly to the inverse of the distance. The constants of this linear relation are determined through a single camera calibration procedure. Having computed the spread, the distance of the object is determined from the linear relation.

In the last class, auxiliary devices are used for depth perception. One of such methods uses a laser pointer which three LEDs are placed on its optical axel [6], built in a pen-like device. When a user scans the laser beam over the surface of the object, the camera captures the image of the three spots (one for from the laser, and the others from LEDs, and then the triangulation is carried out using the camera's viewing direction and the optical axel of the laser. The main problem of these methods is the need for the auxiliary devices, in addition to the camera, and consequently the raise of the complexity and the cost.

2 Suggested Method

This new method includes two steps [3]: First, calculating an interpolation function based on the height and the horizontal angle of the camera. Second, using this function to calculate the distance of the object from the camera.

In the first step, named the primitive evaluation phase, the camera is located in a position with a specified height and a horizontal angle. Then from this position, we take a picture from some lines with equal distance from each other. Then, we provide a table in which the first column is the number of pixels counted from each line to the bottom edge of the captured picture (as the origin line), and the second column is the actual distance of that line from the camera position.

Now, by assigning an interpolation method (e.g. Lagrange method) to this table, the related interpolation polynomial is calculated [2]:

$$f(x) = \sum_{j=0}^{n} f(x_j) L_j(x)$$

$$L_j(x) = \frac{\prod_{i=0, i \neq j}^{n} (x - x_i)}{\prod_{i=0, i \neq j}^{n} (x_j - x_i)} \qquad (1)$$

In this formula, **x** is the distance of the object from the camera, and **n** is the number of considered lines in the evaluation environment in the first step.

In the second step of this method - with the same height and horizontal angle of the camera - the number of the pixels between the bottom edge of the target in the image (the nearest edge of an object in the image to the base of the camera) and the bottom edge of the captured image is counted and considered as **x** values in the interpolation function.

The output of this function will be the real distance between the target in the image and the camera.

This method has some advantages in comparison to the previous methods:

a) Using only a single camera for the depth finding.

b) Having no direct dependency on the camera parameters like focal length and etc.

c) Having uncomplicated calculations.

d) Requiring no auxiliary devices.

d) Having a constant response time, because of having a fixed amount of calculations; so it will be reliable for applications in which the response time is important.

e) The fault of this method for calculating points' distance situated in evaluation domain is too lower.

f) This method can be used for both stationary and moving targets.

However, this method has some limitations such as:

a) The dependency on the camera height and horizontal angle, so that if both or one of them is changed, there will be a need to repeat the first step again.

b) The impracticality of this method for determining the distance of the objects situated out of the evaluation environment (which have been done in the first step).

It is proved that Lagrange Interpolation Method is better than Divided Difference of Newton [4].

3 The Result of Experiment

In this experiment, some lines are drawn on a uniform surface with 5 cm distance from each other. Then a camera is mounted in a position with 45 cm height and 30° horizontal angles.

X	34	64	92	114	136	155	173	189	204	218
Y	5	10	15	20	25	30	35	40	45	50
X	232	245	257	268	279	288	297	304	311	319
Y	55	60	65	70	75	80	85	90	95	100

Table 1: X is the number of pixels between these lines and the origin line in the captured image and Y is actual distance of lines from camera.

Based on counting the pixels between the image of these lines and the origin line (bottom edge of picture) and considering their actual distance, Table No. 1 has been produced:

Using this table and the Lagrange interpolation formula, a function for distance measurement is defined. Then the distance of some random point is calculated with this function as the following table:

Calculated Distance	36.53	60.78	86.18
Actual Distance	36.5	60.9	85.8
Fault percent	0 %	0.20 %	0.44 %

Table 2: Comparison between actual and calculated distance.

As it is realized, this method has more accuracy for measuring the distance of points laid on the primitive environment domain, but out of this domain it is impractical. Considering the properties of this method, it can be used in depth finding systems which have a specified domain, such as the defended systems that react to moving objects in a definite field.

Using this method has no depth limitation provided that the primitive evaluation environment is properly defined. It is needless to say that for increasing the accuracy of the results, the number of lines in the primitive evaluation should be increased.

4 Fault Tolerance in This Method

It does not need to say that one of the important properties that every method must have it, is Fault Tolerance. In the older methods, this issue did not considered at all or because of approximation in most of them, these methods did not have it efficiently.

Here, first we want to deal with problems that cause some faults and then see the effect of these faults on suggested method.

Consider that we have mounted a system on a vehicle based on this method. As this vehicle move along the road, it maybe has some tremors and so has some faults. In other case consider one who wants to capture a picture and estimate the distance upon this captured image. If he does not have enough accuracy to keep the camera along the horizontal line, it has some faults.

For understanding more about fault effect in this method, see the table below for the case that camera has stated with a 16° angle from horizontal line:

Counted pixels	214	280	322
Estimated pixels	41.43	67.45	92.44
Actual pixels	41.2	66.7	91.0
Fault percent	0.56%	1.13%	1.58%

Table 3: Actual and calculated distance and Fault Percent with 16° angle.

If we calculate an interpolation function for the above case and then when we capture an image this angle increased to 18° we have a large Fault Percent as the fallowing table:

Counted pixels	186	252	298
Estimated pixels	31.26	55.44	76.57
Actual pixels	41.2	66.7	91.0
Fault percent	24.1%	16.9%	23.7%

Table 4: Actual and calculated distance and Fault Percent with 18° angle based on previous Interpolation Function.

But we can decrease the effect of this fault with a triangulation method easily.

Now, the way in which Fault Tolerance issue is implemented is explained:

Consider the main angle is **a** and changed in fault case to **b**. According to the following figure we have:

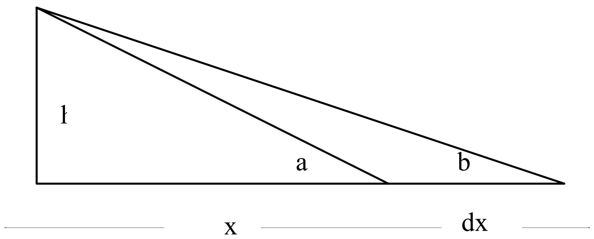

Fig. 1 Fault Tolerance in proposed method

tg(a) = h/x (2)

tg(b) = h/(x+dx)

And then:

x*tg(a) = x*tg(b) + dx*tg(b) (3)

And last:

dx = x*(tg(a) - tg(b)) / tg(b) (4)

that in our example it will be:

dx=0.21748*x (5)

Now, if we correct the estimated distances in Table 4 we have:

X: Previous estimated distance	31.26	55.44	76.57
X+dx: Corrected distance	38.06	67.49	93.22
Actual distance	41.2	66.7	91.0
Fault distance	7.62%	1.18%	2.44%

Table 5: Fault Tolerance decreases the Fault Percent.

As it seems, fault percent decreased to an acceptable range by using this fault tolerance method. So, it can be added to proposed method as a complementally part.

5 Conclusion

The introduced method has some advantages such as simplicity, accuracy, needing no auxiliary devices and no dependency on the camera parameters, compared with the previous methods.

The limitation of this method is the dependency on primitive height and horizontal angle. But the effect of changing these items isn't considerable, and there are some ways to decrease the effect of these faults.

It has also been proved that the Lagrange interpolation method's efficiency is better than the Newton one's in this method.

At last, this method has a good Fault Tolerance Rate.

References:

[1] H. Guo and Y. Lu, "Depth Detection of Targets in a Monocular Image Sequence", *18th Digital Avionic Systems Conference*, 1999.

[2] J. M. Mathews, "*Numerical Methods for Mathematics, Science and Engineering*", Prentice-Hall, 1992.

[3] M. Mirzabaki and A. Aghagolzadeh, "Introducing a New Method for Depth Detection by Camera using Lagrange Interpolation", *The Second Iranian Conference On Machine Vision, Image Processing & Applications*, 2003.

[4] M. Mirzabaki, "Depth Detection Through Single Camera", *The 12th WSCG conference, Czech Republic*, 2004.

[5] M. Subbarao and N. Gurumorthy, "Depth Recovery from Blurred Edges", In *Proceeding of IEEE International Conference on Computer Vision and Pattern Recognition*, 1988.

[6] M. Takatsuka and et al., "Low-Cost Interactive Active Monocular Range Finder", *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1999.

[7]. N. Yamaguti, Sh. Oe and K. Terada, "A Method of Distance Measurement by Using Monocular Camera", *The 36th SICE Annual Conference*, 1997.

[8] Y. L. Murphy and et al., "Depth Finder: A Real-time Depth Detection System for Aided Driving", *IEEE Intelligent Vehicles Symposium*, 2000.