Distance Calculation from Single Optical Image

Xiao-ying DUAN¹,², Yang-jie WEI¹,²,* and Ke ZHANG¹

¹College of Computer Science and Engineering, Northeastern University, Wenhua Str.3, 110819, Shenyang, China
²State Key Laboratory of Synthetical Automation for Process Industries

*Corresponding author

Keywords: Distance calculation, Single image, Oblique straight line, Slope.

Abstract. Accurate and fast distance calculation with respect to a single optical image is useful for real-time 3D construction and acquisition, however currently rare distance calculation methods theoretically base on an optical single image, and the traditional distance calculation method with a single image has limitations due to assumption that the step edge in the image must be strictly horizontal or vertical, which is difficult to fulfill in real applications because the slope of a practical edge could be any other value except for zero and infinity. In this paper, a distance calculation method with a single defocused image containing an oblique step edge is proposed, and no special camera equipment or unique external condition is required. First, the basic theory of coordinate system rotation has been introduced to simplify the sampling process of neighbor points and eliminate the influence resulted from slope of the step edge; one-dimension point-spread-function of the original distance calculation method is expended to two-dimension, taking into account coordinate transform, and a comprehensive distance calculation based on an oblique step edge with any slope is deduced; The relationship between the precision of distance calculation and the slope of the step edge is analyzed and proved in theory. Finally, a serial of simulation are conducted to validate our proposed distance calculation method, and the experimental results prove the applicability, validity and high precision of our method.

Introduction

Real-time distance acquirement with respect to optical images is an important research field in many applications, such as unmanned surface vehicles (USVs), environmental monitoring, and medical operations, to realize autonomy of the used systems.

There are many methods to estimate distance or depth information from 2D optical images, including depth from stereo (DFS), depth from focus (DFF), depth from defocus (DFD) and so on. Compared to DFS and DFF, DFD requires less images, no match and occlusion between images, and the operation is simple. The traditional DFD methods require at least two defocused images with different camera parameters to calculate depth information using the relationship between burring degree and depth [1-4]. Therefore, two cameras with different camera parameters or a camera with two parameter adjustment sets are required. However, it is difficult to modify the camera parameters during image capturing when a real system is operating. Even if two or more images are obtained by two cameras, a time alignment is required to assure that these defocused images are captured at the same time. Therefore, traditional DFD methods are difficult to be used in real-time distance calculation.

In order to calculate distance for real-time applications, to simplify traditional DFD methods in means of improvement on hardware or software is widely researched in recent years. In 1996, Nayar proposed a depth reconstruction method based a single optical image with scene lighting assistance [5], and in 2014, Lin proposed the coded aperture method to modify the camera lens [6]. However, these methods require to improve the hardware or add external settings, therefore their application is greatly limit. In recent years, a new method was proposed by Zhou and Cao [7-9], in which only a
defocused image is captured and the second defocused image is synthetized with a known defocus degree as expected, then the distance information in the original image is calculated based on traditional DFD fundamentals. However, this method requires an additional step to synthetize the second defocused image after the first defocused image is captured, therefore theoretically it is not a distance reconstruction method with a single image, and of course it still can not be used in a real-time dynamic application.

Distance calculation from a single optical image with a step edge is first proposed by Pentland in 1987[5]. This method estimates the blurring degree by comparing the intensity distribution on both sides of a step edge, and obtains distance between the edge and the camera with the relationship between blurring degree and distance. The algorithm is based on an assumption that the step edge in the image is horizontal or vertical. However, in distance calculation of real systems whose application environment is complicated, it is difficult to assure that the step edge in a dynamic image is strictly vertical or horizontal. Therefore, the application fields of this original method are limit.

In order to calculate distance information with a single optical image captured by a real-time system, the paper expands the theories of the original distance calculation method and solves the problem when the step edge is oblique and the point-spread-function is two-dimensional spreading, and then proposes an improved distance calculation method based on a single image where the step edge could be positioned at different positions and with different slopes. Compared to the original method, our method needs only an image with a step edge to calculate the distance information between the edge and the camera, and no assumption about the step edge is required.

**Distance Calculation with a Single Image**

**Imaging Model**

In optical imaging, the object distance u of a source point \( P_0 \), the image distance v and the focal length f meet the Gauss imaging formula shown as Eq. (1), the image of \( P_0 \) is a point, which is also called focused image. If any parameter in Eq. (1) is changed, the image of \( P_0 \) becomes a blurred round spot, called as defocused image.

\[
\frac{1}{u} + \frac{1}{v} = \frac{1}{f}
\]

(1)

According to the principle of light propagation, the intensity distribution function in the spot is called as point-spread-function (PSD), which is approximately equal to a two-dimensional Gauss function,

\[
h(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2 + y^2}{2\sigma^2}}
\]

(2)

where \( x \) and \( y \) respectively represent the horizontal and vertical coordinates of point \( (x, y) \) on the imaging plane; \( \sigma \) is the variance of Gauss function, which is used to represent the blurring degree.

Theoretically, a defocused image can be considered as the summation of all round spots, therefore it can be expressed as,

\[
G(x, y) = h(x, y) \ast I(x, y)
\]

(3)

where \( G(x, y) \) represents the defocused image; \( I(x, y) \) represents the focused image; “\( \ast \)” represents convolution.

According to the geometry knowledge, the distance between \( P_0 \) and the lens can be calculated with,

\[
d = \frac{fv}{v - f - m\sigma F}
\]

(4)
where \( m \) represents the coefficient between \( r \), radius of the defocused spot, and \( \sigma \); \( F \) is the aperture number of the camera.

From Eq. (4), it can be seen that with fixed camera parameters, \( d \) is easily calculated if the blurring degree \( \sigma \) is estimated precisely. In order to obtain \( \sigma \) of each pixel in a defocused image, either its focused image is already known, or another defocused image of the identical scene with different camera parameters is used to be compared to. However, in a dynamic application, it is difficult to fulfill these requirements because there is no time to vary camera parameters and it is impossible to capture a defocused image in an outdoor environment. If there is a step edge in the defocused image, the original distance calculation method proposed by Pentland can be used to calculate distance information of pixels on this edge, because the blurring degree of each point can be estimated through estimating the intensity variation of both sides of the edge. The basic theory is denoted as following.

**Original Distance Calculation with a Single Image**

Suppose that the defocused image has a vertical step edge at point \( x_0 \) shown as Fig. 1, the intensity of the step edge can be represented as,

\[
I(x, y) = \begin{cases} 
  k & \text{if } x-x_0 < 0 \\
  k + \delta & \text{if } x-x_0 \geq 0 
\end{cases}
\]

where \( k \) is the intensity value of point \((x, y)\); \( \delta \) represents the intensity difference.

![Figure 1. The diagram of a vertical edge in an image.](image)

Taking the Laplace transformation in Eq. (3), we could obtain,

\[
L(x, y) = \nabla^2 \left( h(x, y) \ast I(x, y) \right) = \nabla^2 \left\{ \int \int \frac{1}{2\pi\sigma^2} e^{-\frac{(x-u)^2}{2\sigma^2}} I(u, v) \, du \, dv \right\}
\]

where “\( \nabla \)” is Laplace transformation; \( L(x, y) \) represents the rate of intensity-variation at point \((x, y)\).

When we estimate the blurring degree of the step edge, we only need to compare the intensity difference between points on two sides of the edge, while the intensity distribution of the pixels along the step edge does not need to be considered. Therefore, the PSD in Eq. (2) can be simplified into an unidirectional function, and Eq. (6) is transformed into,

\[
L(x, y) = -\frac{\delta}{\sqrt{2\pi\sigma^2}} \int_{x_0}^{\infty} \left( 1 - \frac{(x-u)^2}{\sigma^2} \right) e^{-\frac{(x-u)^2}{2\sigma^2}} du = -\frac{\delta}{\sqrt{2\pi\sigma^2}} (x-x_0) e^{-\frac{(x-x_0)^2}{2\sigma^2}}
\]

Taking the absolute and the natural logarithm on both sides of Eq. (7), we obtain,

\[
\ln \frac{\delta}{\sqrt{2\pi\sigma^2}} (x-x_0) = \ln \left| \frac{L(x, y)}{x-x_0} \right|
\]

which can be simplified as,

\[
A(x-x_0)^2 + B = C(x-x_0)
\]
where \( A = -\frac{1}{2\sigma^2} \), \( B = \ln \frac{\delta}{\sqrt{2\pi}} \), \( C = \ln \frac{L(x,y)}{|x-x_0|} \)

As a linear regression problem about \( x^2 \), the coefficients \( A \) and \( B \) can be obtained using maximum likelihood estimation,

\[
A = \frac{\sum [(x_i - x_0)^2 - (x-x_0)^2]}{\sum [(x_i - x_0)^2]}, \quad B = \bar{C} - \left(\frac{x-x_0}{2}\right)^2
\]

(10)

where \( x_i \) is the horizontal coordinate of neighbor points of \( x_0 \) which is on the step edge; \( x_i-x_0 \) is the distance between the sample point \( x_i \) and \( x_0 \), \( x-x_0 \) represents the average distance between sample points and \( x_0 \); \( \bar{C} \) represents the average of \( C \).

Then we can calculate the blurring degree with,

\[
\sigma = \frac{1}{\sqrt{-2A}}
\]

(11)

Finally, replace \( \sigma \) in Eq. (4), the distance information of each point on the step edge can be obtained,

\[
d = \frac{fy}{v-f-m\gamma\sqrt{-2A}}
\]

(12)

If the step edge is horizontal, the deduction process is similar and simple because only \( x \) is replaced by \( y \). However, it is difficult to use this distance calculation method in real applications due to the assumption that the step edge is vertical or horizontal, which is difficult to assured. Therefore, in this paper an improvement to this original method has been proposed to expand its application areas where the step edge is oblique.

**Distance Calculation of Oblique Step Edges**

There is an oblique step edge in the defocused image shown as Fig.2, the intensity value of the edge can be represented as,

\[
I(x,y) = \begin{cases} 
  k & x-x_0-t\gamma < 0 \\
  k+\delta & x-x_0-t\gamma \geq 0 
\end{cases}
\]

(13)

where \( 1/\gamma \) is the slope of the step edge and \( \tan\theta=1/\gamma \).

In order to calculate distance, first the original coordinate system is rotated \( \alpha \) degrees until the new vertical axis \( y' \) is parallel to the step edge. According to the theory of coordinate system rotation, the new coordinates \((x', y')\) can be denoted by the old coordinates \((x, y)\) and a rotation matrix, which is show as,

\[
\begin{bmatrix}
  x' \\
  y'
\end{bmatrix} = \begin{bmatrix}
  \cos(-\alpha) & \sin(-\alpha) \\
  -\sin(-\alpha) & \cos(-\alpha)
\end{bmatrix} \begin{bmatrix}
  x \\
  y
\end{bmatrix}
\]

(14)

\[\alpha = \arctan(\gamma)\]

(15)

According to Eq. (14), the new coordinates of point \((x', y')\) can be denoted as,

\[
x' = x \cos \alpha - y \sin \alpha \\
y' = x \sin \alpha + y \cos \alpha
\]

(16)

Therefore, after rotation, the intensity value of the step edge in Eq. (13) can be represented as,
From Eq. (17), it can be seen that after coordinate system rotation the influence of the slope $t$ has been eliminated and all the calculation equations in original method could be used here.

Then, replace $I(x, y)$ in Eq. (6) with $I(x', y')$,

$$L'(x', y') = \sqrt{2} \int \int \frac{1}{2\pi\sigma} e^{-\frac{(x'-x_0)^2 + (y'-y_0)^2}{2\sigma^2}} I(u', v') du' dv'$$

(18)

According to the original method in Section 2, Eq. (18) can be simplified as,

$$L'(x', y') = -\frac{\delta}{\sqrt{2\pi}\sigma}(x'-x_0')xe^{-\frac{(x'-x_0')^2}{2\sigma^2}}$$

(19)

where $x_0'$ is equal to,

$$x_0' = x_0 \cos \alpha$$

(20)

Taking the absolute and the natural logarithm on both sides of Eq. (19), we obtain,

$$A'(x'-x_0')^2 + B' = C'(x'-x_0') \quad A' = -\frac{1}{2\sigma^2} \quad B' = \ln \frac{\delta}{\sqrt{2\pi}\sigma} \quad C' = \ln \left| \frac{L'(x', y')}{{x'-x_0'}} \right|$$

(21)

$$L'(x', y') = L(x \cos \alpha - y \sin \alpha, x \sin \alpha + y \cos \alpha)$$

(22)

$$A' = \frac{\sum((x_i'-x_0')^2-(x'-x_0')^2)}{\sum((x_i'-x_0')^2-(x'-x_0')^2)} \quad B' = C' - (x'-x_0')^2$$

(23)

where $x_i'$ is the horizontal coordinate of neighbor points of $x_0'$ in the new coordinate system; $x_i' - x_0'$ is the distance between the sample point $x_i'$ and $x_0'$; $\overline{x_i'-x_0'}$ represents the average distance between the sample point $x_i'$ and $x_0'$; $\overline{C_i'}$ represents the average of $C_i'$.

Replace $x', y'$, and $x_0'$ in $A'$ and $L'$ of Eq. (22), then,

$$A = \frac{\sum(x_i \cos \alpha - y_i \sin \alpha - x_0 \cos \alpha - y_0 \sin \alpha)^2 - \sum(x_i - x_0)^2 \sum(x_i' - x_0')^2}{\sum(x_i - x_0)^2 \sum(x_i' - x_0')^2} \quad C_i = \ln \left| \frac{L(x_i \cos \alpha - y_i \sin \alpha, x_i \sin \alpha + y_i \cos \alpha)}{x_i \cos \alpha - y_i \sin \alpha, x_i \sin \alpha + y_i \cos \alpha} \right|$$

(24)

Replace $\alpha$ with $t$ in Eq. (24), then $A'$ can be calculated with $t$. Subsequently, the blurring degree $\sigma$ can be calculated with coordinates in the original coordinate system with Eq. (11) and Eq. (24). That means even the step edge is oblique, its distance to the camera can be calculated with our method and the calculation equation is denoted as,

$$d = \frac{fv}{v-f-mF\sqrt{2A}}$$

(25)

If $t=0$, $\alpha = 0^\circ$, then,

$$A' = \frac{\sum((x_i - x_0')^2 - x - x_0^2)}{\sum((x_i - x_0')^2 - x - x_0^2)} = A \quad C_i = \ln \left| \frac{L(x, y)}{x - x_0} \right| = C_i$$

(26)

From Eq. (26), it can be seen that the original distance calculation method is only a special case of our proposed method; when the step edge is not oblique, the calculation result of our method and the original method is the same. In the following, the influence of the slope $t$ on the distance calculation result is analyzed.
First, it is reasonable to assume that when we estimate variation of blurring degree, the number of sample points \((x_i, y_i)\) on two sides of the step edge is the same, thus the average distance between the sample points and the step edge is zero. That means \((x_i \cos \alpha - y_i \sin \alpha - x_0 \cos \alpha)^2 = 0\). Furthermore, \(L\) is the Laplace transformation of the image, and it does not influenced by coordinate system rotation, thus it can be taken as a constant \(T\).

Therefore, Eq. (24) can be simplified as,

\[
A' = \frac{\sum z_i^2 \ln \left| \frac{z_i}{z_0} \right|}{\sum (z_i^2)}
\]  

(27)

where \(z_i = x_i \cos \alpha - y_i \sin \alpha - x_0 \cos \alpha\).

Then,

\[
A' = \frac{\sum [z_i^2 (\ln |T| - \ln |z_i|)]}{\sum (z_i^2)}
\]  

(28)

When the rotation angle \(\alpha\) increases from 0 to \(\pi/2\), \(z_i\) is inversely proportional to \(\alpha\). In detail, the denominator of Eq. (28) decreases with the speed of the \(z_i^4\), while the numerator decreases with the speed of the \(z_i^2 \ln |z_i|\). Because the molecules decrease faster than the denominator in Eq. (28), \(A'\) is proportional to \(\alpha\). According to Eq. (11), \(\sigma\) is inversely proportional to \(\alpha\) and \(1/t\). Therefore, when the angle \(\Theta\) between the step edge and the horizontal axis is equal to 90°, the calculation result of our method and the original method is the same. However, when \(\Theta\) decreases, \(\sigma\) also decreases and \(A'\) is larger than that of the original method which has not considered the influence of \(t\). That means when the angle of the step edge between the vertical axis and the edge becomes smaller, the calculated distance of the original method is larger than that of our method. Therefore, with decreasing of \(\Theta\), the precision of the original method becomes much lower, while the precision of our method will not influenced by the slope of the step edge.

**Simulation**

In order to validate our proposed algorithm of distance calculation, first, a serials of simulation are conducted. The camera parameters are as follows: the ideal object distance is 850mm; the focal length is 1.2mm; the radius of the lens is 0.6mm; the aperture number is 2.0. The defocused images consists of a rectangle box on a flat substrate is synthetized; the bottom and top surface of the box is flat, and the height of the box is 800mm; the distance between the substrate and the camera is 850mm. Since the distance between the top surface and the camera is less than the ideal object distance, the synthesized image of the top surface is defocused, while the image of the substrate is focused. In our simulation, the distance between the camera and the top edge of the box is calculated with the original method and our method. From the simulation condition, it can be seen that the ground truth is 500mm. The diagram of our simulation is shown in Fig. 3. In order to test the robustness of our method, we add the different brightness levels on the rectangle box and the substrate of the synthesized image.

![Figure 3. The diagram of our simulation.](image)

![Figure 4. Defocused image of oblique edge with slope of 0.8.](image)
First, the synthesized defocused image composed of a box and a substrate is shown in Fig. 4, where the slope of the long top edge of the box is 0.8. Then, the distance between the top edge and the camera is calculated with the original method and our method in this paper. The calculation result is shown in Fig. 5, where the horizontal axis denotes the pixel number on the edge; the vertical axis is distance with unit of m; the line with “*” is the true distance value; the line with “o” is the distance calculated with the original method; the line with “+” is the distance calculated with our method in this paper.

![Image of Fig. 5](image1)

![Image of Fig. 6](image2)

Figure 5. Distance calculation result with slope of 0.8. Figure 6. Distance calculation result with slope of 3.0.

From Fig. 5, it can be seen that when the slope of the step edge is 0.8, which is far from the vertical edge, for most of points the calculated distance of our method is closer to the ground truth than that of the original method.

In order to prove the precision of our method, the mean-relative-error (MRE) and the mean-square-deviation (MSD) of calculated distance are calculated with the following equations,

\[
MRE = \frac{\sum_{i=1}^{n}|d_i - d|}{nd}
\]

(29)

where \(d\) is the true distance; \(d_i\) is the distance of the pixel \(i\); \(n\) is the number of pixels along the step edge.

\[
MSD = \sqrt{\frac{\sum_{i=1}^{n}(d_i - d)^2}{n}}
\]

(30)

With Eq. (29) and Eq. (30), MRE and MSD are calculated based on the result in Fig. 5. MRE of the original method is 0.0542, and MRE of our method is 0.0090; MSD of the original method is 0.2354, MSD of our method is 0.0363. Therefore, MRE and MSD of our method are both lower than those of the original method. That means the precision of our method is higher than the original method when the defocused image has an oblique edge.
Second, a series of similar simulations are conducted with respect to different oblique step edges to research the relationship between slope and precision of distance calculation. In our simulation, the slope of the oblique edges is 3.0, 2.0, and 1.0, respectively. The distance of the top edge is calculated with the original method and our method, and the calculation result is shown in Figs. 6-8. Based on these figures, we also calculate the error value of each point for both the original method and our proposed method.

From Figs. 6-8, it can be seen that the precision of the original method is much easier to be influenced by the slope of the step edge, while the precision of our method isn’t obliviously influenced by it. In order to compare the influence of slope on the original method and our method methods in detail, MRE and MSD of the original method and our method are calculated with respect to different step edges, and the result is shown in Table 1 and Table 2.

| Method        | slope 0.8 | slope 1.0 | slope 2.0 | slope 3.0 |
|---------------|-----------|-----------|-----------|-----------|
| Original method | 0.0542    | 0.0388    | 0.0297    | 0.0202    |
| Our method     | 0.0090    | 0.0087    | 0.0094    | 0.0099    |
| Difference     | 0.0452    | 0.0301    | 0.0203    | 0.0103    |

| Method        | slope 0.8 | slope 1.0 | slope 2.0 | slope 3.0 |
|---------------|-----------|-----------|-----------|-----------|
| Original method | 0.2354    | 0.2063    | 0.1819    | 0.1435    |
| Our method     | 0.0363    | 0.0354    | 0.0300    | 0.0344    |
| Difference     | 0.1991    | 0.1709    | 0.1519    | 0.1091    |

From Tables 1-2, the following conclusion can be obtained,

1) When the slope of the step edge decreases from 3.0 to 0.8, MRE of the original method increases from 0.0202 to 0.0542, while MRE of our method is 0.0099 when the slope is 3.0 and it does not vary much with the variation of slope.

2) When the slope of the step edge decreases from 3.0 to 0.8, MSD of the original method increases from 0.1435 to 0.2354, while MSD of our method is 0.0344 when the slope is 3.0 and it does not vary much with the variation of slope.

3) When the step edge is close to be vertical, the precision of the original method is similar to that of our method; however when the step edge is further to be vertical, the precision of the original method is much lower, while the precision of our method has not changed.

4) The average MRE and MSD of our method are 0.00925 and 0.0340, respectively. While the average MRE and MSD of the original method are 0.0357 and 0.191775, respectively.
Summary
In this paper, distance calculation with a single defocused image containing an oblique step edge without special camera equipment or unique external conditions is proposed. The first contribution is to introduce coordinate system rotation to fulfill the requirement of original distance method and eliminate the influence of slope; The second contribution is enlarging point-spread-function from one-dimension to two-dimension, taking into account coordinate transform, and a comprehensive distance calculation based on an oblique step edge with any slope is deduced; The relationship between the precision of distance calculation and slope of step edges is analyzed with both MRE and MSD of distance calculation. Finally, a serial of simulation is conducted to validate the distance calculation method proposed in this paper, and the experimental results prove the applicability, validity and high precision of our method.

Acknowledgement
This research was financially supported by the National Key Research and Development Plan (2016YFC0101500) and the Fundamental Research Funds for the Central Universities (N161602002) and State Key Laboratory of Synthetical Automation for Process Industries.

References
[1] Pentland A.P. A New Sense for Depth of Field [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1987, pp. 523-531.
[2] Subbarao M. Parallel Depth Recovery By Changing Camera Parameters [C]. Computer Vision, 1988, pp.149-155.
[3] Namboodiri V.P, Chaudhuri S. On defocus, diffusion and depth estimation [J]. Pattern Recognition Letters, 2007, pp. 311-319.
[4] Wei Y.J, Dong Z.L, Wu C.D. Global depth from defocus with fixed camera parameters [C]. International Conference on Mechatronics and Automation. 2009, pp. 1887-1892.
[5] Nayar S.K, Watanabe M, Noguchi M. Real-time focus range sensor [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1996, Vol. 18(12), pp. 1186-1198.
[6] Lin J.Y, Lin X, Ji X.Y, Dai Q.H. Separable Coded Aperture for Depth from a Single Image [J]. IEEE Signal Processing Letters, 2014, Vol. 21(12), pp. 1471-1475.
[7] Zhou S., Sim T. On the recovery of depth from a Single defocused image [C]. Proceedings of the 13th International conference Computer Analysis of Images and Patterns, 2009, Vol. 57(2), pp. 889-897.
[8] Ming Y., Jiang J.J. Depth recovery from a single defocused image using a Cauchy-distribution-based point-spread-function model [J]. Journal of Image and Graphics, 2015, Vol. 20(05), pp. 708-714.
[9] Cao F.Y., Fang S., Hu Y.J., Wang H., Yang J. Recovering depth from a single natural defocused image [J]. Journal of Image and Graphics, 2014, Vol. 19(05), pp. 746-754.