ELECTRONICS AND ELECTRICAL ENGINEERING
ISSN 1392 – 1215 2008. No. 8(88)
ELEKTRONIKA IR ELEKTROTECHNIKA

T170 ELECTRONICS
ELEKTRONIKA

Subpixel Edge Reconstruction using Aliased Pixel Brightness

V. Vyšniauskas
Šiauliai University,
Vilniaus str. 141, LT – 76353 Šiauliai, Lithuania

Introduction

One of the most common image features used in machine vision are edges, and there is a substantial body of research on various techniques for performing edge detection. Edge is an imaginable line that separates two regions with different luminosity. When luminosity hops sharply, edge is well visible or imaginary, but when change of luminosity is slight – edge is very light or nearly invisible.

There are many methods for edge detection, but most of them can be grouped into two categories, zero-crossing and search-based. The search-based edge detects methods looking for maximum and minimum in the first derivative of the image. The zero-crossing based methods search for zero crossings in the second derivative of the image [1].

A drawback with using edges is that not only do edge detectors but also extract meaningful and useful edges. In addition, many other spurious ones that arise from noise and small changes in intensity values. If all such edges are kept then the resulting image is hard for subsequent processing. Large number of edge points can seriously increase computational amount and decrease result quality with remaining fraction of noise. The alternative is to select a subset of edges for further analysis and ignore the remaining fractions. Generally, a threshold on the gradient magnitude of pixels solves this problem. Unfortunately, in practice edge thresholding often done intuitively and frequently requiring user tuning of parameters [2]. Higher threshold level usage results lost of some necessary edges contrariwise lower threshold level leaves more unnecessary fractions. Optimal threshold level defined for each image individually sometimes separately for different parts of the same image [2, 3].

Image projection and digitizing

Image cameras have a lens to gather the incoming light and focus all or part of the image on the image sensor surface. Image sensor is flat panel, with sensitive to light elements. Image sensors grouped in two categories, analogue and digital. Analogue sensors output analogue signal, which is used in analogue television or digitalized to obtain digital image. Digital image sensors are made from millions square shape light sensors that are capturing light and converting it into electrical signals. Sensors are organized in rows and columns as rectangular matrix. One light sensor represents a single dot in the image with some luminosity and are named pixel. Pixel of gray scale image represents with one brightness digit, each pixel of colour image represented as three colors – red, green and blue brightness digits. Most popular quantization is 8 bits or 1 byte but also are used 10 or more bits quantization. Image quality depends on quantization directly.

Each pixel of such image is neither a dot nor a square but an abstract sample. Pixels could be reproduced at any size and shape as a slight visible dot or a square.

Actually image sensor consists of square shape pixels, which have some micrometers in size. Each pixel of image has a fixed position and variable brightness which represents average luminosity on pixel area (Fig. 1). Black diagonal line shows edge between light and dark areas.

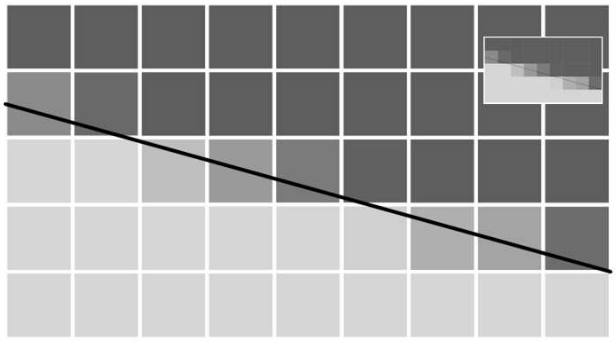

Fig. 1. Pixel aliasing phenomenon

Therefore, a projected image light and dark part covers some pixel surface partially. These pixels obtain average brightness from both light and dark parts. This phenomenon is named pixel aliasing [4] and presents in grayscale and color images. Aliased pixel brightness depends on light and dark areas ratio and luminosity of these areas.

It is evident that this pixel brightness is directly proportional to dark and light area ratio and difference of these areas luminosity is a rate factor.

Edge reconstruction from aliased pixel brightness

Decision about aliased pixel brightness is described as formula (1).

$$B_P = \left((B_L - B_D) \cdot \frac{S_L}{S_P} \right) + B_D \,, \tag{1}$$

where B_P – average (visible) brightness of pixel, B_L – brightness of light area, B_D – brightness of dark area, S_P – whole pixel area, S_L – pixel light area.

Brightness of the light B_L and dark B_D areas is obtainable from nearest neighbour pixels in opposite sides of aliased pixel.

Light area S_L is an integral of commonly unknown function of an edge $y = f(x)$.

$$S_L = \int_0^1 f(x)dx. \tag{2}$$

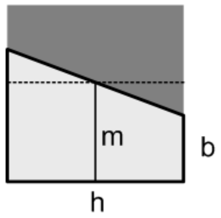

Fig. 2. Trapezium area

To simplify task, let's replace function segment with line (Fig. 2). It is possible because pixel is a smallest peace of the image represented as a dot or a square with monotonous brightness. Yet another assumption that edge line cross two opposing sides of pixel. Assumed the decisions light area S_L is calculated as trapezium.

$$S_L = h \cdot (a+b)/2 \tag{3}$$

or

$$S_L = h \cdot m \,, \tag{4}$$

where $m = a + b/2$ is average of vertical length of the trapezium. Pixel side length is marked h. Pixel light area and a whole pixel area ratio is the same as *m* and *h* ratio.

$$\frac{S_L}{S_P} = \frac{m}{h}. \tag{5}$$

Value *m* can be formulated from formulas (1) and (5)

$$m = h \cdot \left(\frac{B_P - B_D}{B_L - B_D} \right). \tag{6}$$

To reconstruct edge position with subpixel accuracy four steps must be done: 1) to find aliased pixel; 2) to select two nearest opposite pixels with highest and lowest brightness; 3) to calculate *m* line value by formula (6), that is a distance from brightest pixel border to estimated dot on edge; 4) to draw a line through these dots that estimates real edge with subpixel accuracy (Fig. 3).

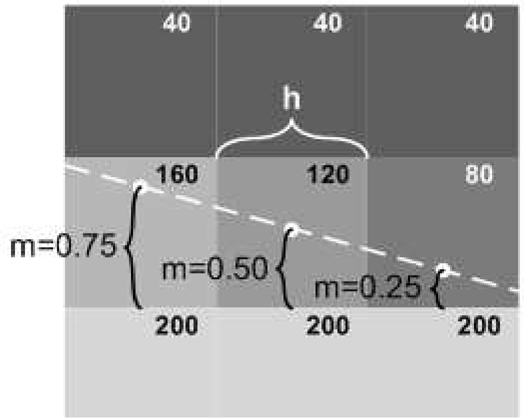

Fig. 3. Subpixel edge dots calculation

$$m_1 = 1 \cdot [(160-40)/(200-40)] = 0.75;$$
$$m_2 = 1 \cdot [(120-40)/(200-40)] = 0.50;$$
$$m_3 = 1 \cdot [(80-40)/(200-40)] = 0.25.$$

Calculation is illustrated in (Fig. 3). Here 9 pixels are shown and edge (dash and line) goes through a middle pixel row. Light pixel brightness is 200, dark – 40 and aliased with brightness 160, 120, 80. From pixel brightness calculated distances *m* and these dots used to draw an estimated edge line. The distance *m* are calculated from the pixel brightness and these dots are used to draw estimated edge.

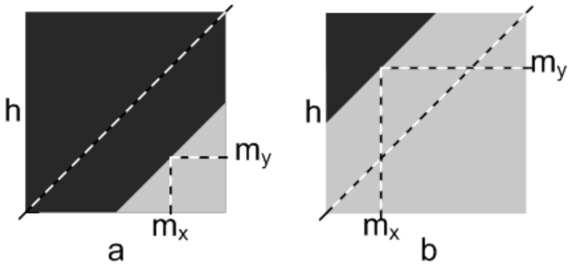

Fig. 4. Edge intercepts triangle area

Different situation is when an edge crosses adjacent sides of pixel and intercepts triangle area. This situation is more complicated because triangle area (Fig. 4) is not linear function when edge dot travels via diagonal from one corner to diagonally opposite corner. This area also depends on angle between edge and pixel horizontal or vertical side. That additionally complicates problem decision.

To simplify the task take assumption that edge is parallel to pixel diagonal, then $(m_x = m_y)$.

Two different formulas (7) are used to calculate estimated edge dot coordinates m_x, m_y. The first one (upper formula) calculates coordinates when a dot is in the first half of pixel (Fig.4a), and the second formula is used to calculate coordinates when pixel is in the second half of pixel (Fig. 4b).

$$m_x = m_y = \left\{ \begin{array}{ll} \dfrac{\sqrt{S_L}}{\sqrt{2}}, & m_x \le \dfrac{h}{2} \\ h - \dfrac{\sqrt{h^2 - S_L}}{\sqrt{2}}, & m_x > \dfrac{h}{2} \end{array} \right\}; \tag{7}$$

where

$$S_L = h^2 \cdot \left(\frac{B_P - B_D}{B_L - B_D} \right) = h^2 \cdot BrR\ , \tag{8}$$

$$BrR = \left(\frac{B_P - B_D}{B_L - B_D} \right). \tag{9}$$

where BrR – brightness ratio.

Final formula is:

$$m = \left\{ \begin{array}{ll} h \cdot \dfrac{\sqrt{BrR}}{\sqrt{2}}, & B_P \le \dfrac{B_L - B_D}{2} + B_D \\ h \cdot \left(1 - \dfrac{\sqrt{1 - BrR}}{\sqrt{2}} \right), & B_P > \dfrac{B_L - B_D}{2} + B_D \end{array} \right\}; \tag{10}$$

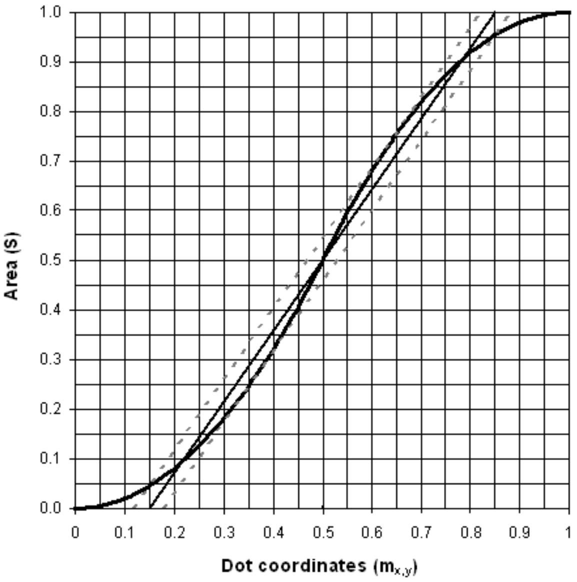

Fig. 5. Area (S) to dot coordinates (m) linearity diagram

Formulas describe the relation of dots coordinates and area size that is an S shape curve. In this function selected region with linearity better than 5%. It is Area (S) range from 0.1 to 0.9 where subpixel dot coordinates (m) range varies from 0.15 to 0.85. It means that, when edge intercepts small triangle that area is less than 0.1 (10%) of pixel size, calculation is inaccurate. Such pixels can be ignored or another calculation algorithm must be used. Edge reconstruction where used both methods shown in Fig. 6.

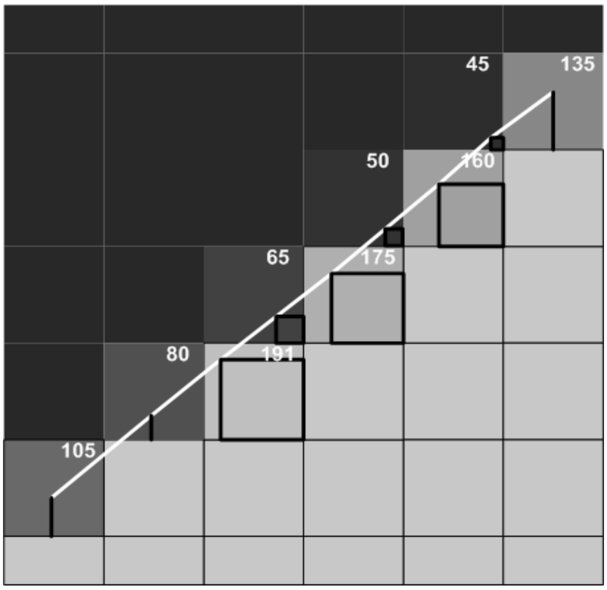

Fig. 6. Both methods for edge reconstruction usage

Testing and results

Artificial pictures with only one straight-line edge between dark and light areas were used for testing. There were used pictures with known different edge angles. This decision was made to simplify testing and results analysis.

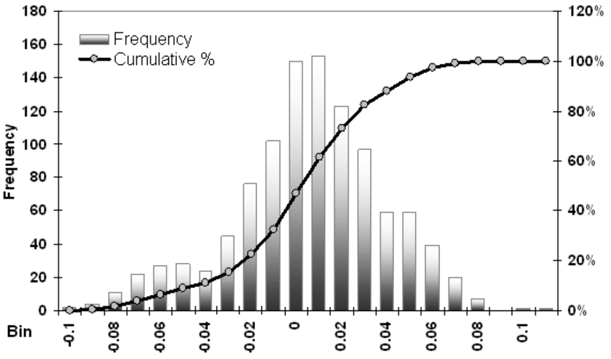

Fig. 7. Edge reconcrtuction deviation histogram

Table 1. Standard deviation cumulative percents

0.01 (1%)	47%	0.02 (2%)	61%
0.03 (3%)	73%	0.04 (4%)	82%
0.05 (5%)	88%	0.06 (6%)	94%

There were tested 14 pictures and calculate 1050 dots. Each calculated dot position was compared with known angle straight-line and calculated deviation. Test results were drawn as histogram of edge reconstruction deviation shown in Fig. 7. Table 1 shows standard deviation cumulative percents Method accuracy is six or less percent.

Conclusion

Well-known edge detectors cany, sobel, perwit and other detect edge from blurred images to reduce noise [5, 6]. Trash-hold level is used to extract edge line. Unfortunately, in practice edge thresholding is often done intuitively and frequently requiring user tuning of parameters. Accordingly, edge line width is of one or more pixels accuracy. These methods are unusable in applications where high accuracy is need.

Aliased pixel contains information about ratio between light and dark areas that have covered pixel. This topic uncovers methods how to get point coordinates with subpixel precision. Presented point of edge estimation precision is about 5 percent (Table 1) of the pixel width. This method does not require Gaussian blur and threshold turning. Edge detection (restoration) precision is a part of pixel.

The main uncovered problem of this method is aliased pixel detection that will be solved in future.

References

1. **Russ John C.** The image processing handbook, 5th ed. – CRC Press Taylor & Francis Group. – 2006. – P. 19–25, 135–145, 292–315.
2. **Ramanauskas Nerijus** The Investigation of Eye Tracking Accuracy using Synthetic Images // Elektronika ir elektrotechnika. – Kaunas: Technologija, 2003. – Nr.4(46). – P. 17–20.
3. **Ritter G. X., Wilson J. N.** Handbook of Computer Vision Algorithms in Image Algebra. – CRC Press. – 1996. – P. 105–121.
4. **Ling Guan, Sun-Yuan Kung, Jan Larsen.** Multimedia image and video processing. – CRC Press LLC. – 2001. – P. 83–111.
5. **Hansen C., Johnson C. R.** The Visualisation Hand Book. – Elsevier Butterworth–Heinemann. – 2005. – P. 150–162.
6. **Nixon M. S., Aguado A. S.** Feature Extraction and Image Processing. – 2002. – P. 99–130.

Received 2008 02 19

V. Vyšniauskas. Subpixel Edge Reconstruction using Aliased Pixel Brightness // Electronics and Electrical Engineering. – Kaunas: Technologija, 2008. – No. 8(88). – P. 43–46.

One of the most common image features used in machine vision are edges, and there is a substantial body of research on various techniques for performing edge detection. Edges are useful in many applications as image comparing, recognition and other. Here is presented edge detection method with subpixel accuracy. Method is based on decision that different intensity and size areas influence pixel brightness with some relation function. Hear presented functions to calculate one dot of edge going through the pixel. Test results show that with 0.01 standard deviation is estimated 47% of dots, with 0.05 standard deviation is estimated 88% of dots and 94% with 0.06 standard deviation. Also it is defined, that linearity decrease is more than 5% when edge cut triangle which area is less then 10% of pixel area. Ill. 7, bibl. 6 (in English; summaries in English, Russian and Lithuanian).

В. Вишняускас. Восстановление контура используя яркость пикселя // Электроника и электротехника. – Каунас: Технология, 2008. – № 8(88). – С. 43–46.

Контур – одна из самых общих характеристик изображения, используемых в машинном зрении. Существует множество различных методов для обнаружения контура. Контур полезен во многих применениях таких как сравнивнение, опознавание изображения и других. Представляется метод обнаружения контура с точностью до доли пикселя. Метод основан на решении, что зоны разной интенсивности и размера влияют на яркость пикселя и представляют некоторую функцию. Представлены функции для вычисления точки контура, находящейся на пикселе. Результаты исследования показывают, что с 0.01 стандартными отступлениями определены 47 % точек, с 0.05 стандартными отступлениями – 88 % точек и 94 % точек с 0.06 стандартными отступлениями. Также определена нелинейность более 5 %, когда контур отсекает треугольник площадью менее 10 % от площади пикселя. Ил. 7, библ. 6 (на английском языке; рефераты на английском, русском и литовском яз.).

V. Vyšniauskas. Vaizdo kontūrų atkūrimas naudojant persidengusių taškų ryškumą // Elektronika ir elektrotechnika. – Kaunas: Technologija, 2008. – Nr. 8(88). – P. 43–46.

Vaizdo kontūro nustatymas yra viena iš bendriausių vaizdų palyginimo, atpažinimo ir kitokio apdorojimo charakteristikų. Vaizdo pakeitimas kontūru leidžia gerokai sumažinti kompiuterio skaičiavimų trukmę. Kontūrams nustatyti naudojami įvairūs metodai Pristatomas metodas vaizdo kontūrui nustatyti pikselio dalies tikslumu. Metodas paremtas tuo, kad skirtingo ryškio sritys, dengiančios tą patį pikselį tam tikru proporcingumu, daro įtaką bendram pikselio ryškumui. Pateikiamos funkcijos per pikselį einančio kontūro taško koordinatėms rasti. Tyrimais nustatyta, kad su 0,01 neapibrėžtimi nustatomi 47 % taškų, su 0,05 neapibrėžtimi – 88 % taškų, o su 0,06 neapibrėžtimi – 94 % taškų. Taip pat nustatyta, kad netiesiškumas viršija 5 %, kai kontūras atkerta trikampį, kurio plotas sudaro 10 % pikselio ploto. Tokius pikselius reikia ignoruoti. Il. 7, bibl. 6 (anglų kalba; santraukos anglų rusų ir lietuvių k.).

DOI: 10.5755/j02.eie.11231