(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
2 June 2016 (02.06.2016)

WIPO | PCT

(10) International Publication Number
WO 2016/083002 A1

(51) International Patent Classification:
G06K 9/00 (2006.01) *G06K 9/46* (2006.01)

(21) International Application Number:
PCT/EP2015/073058

(22) International Filing Date:
6 October 2015 (06.10.2015)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
62/085,346 28 November 2014 (28.11.2014) US

(71) Applicant: **FOTONATION LIMITED** [IE/IE]; Cliona, Building 1, Parkmore East Business Park, Ballybrit, Co. Galway (IE).

(72) Inventors: **MUNTEANU, Mihai Constantin**; 2B Macin Street, R-500133 Brasov (RO). **GEORGESCU, Vlad**; 2B Macin Street, R-500133 Brasov (RO).

(74) Agent: **HANNA MOORE + CURLEY**; Garryard House, 25/26 Earlsfort Terrace, Dublin 2, D02 PX51 (IE).

(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JP, KE, KG, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) Designated States *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

Published:

— *with international search report (Art. 21(3))*

(54) Title: A METHOD FOR PRODUCING A HISTOGRAM OF ORIENTED GRADIENTS

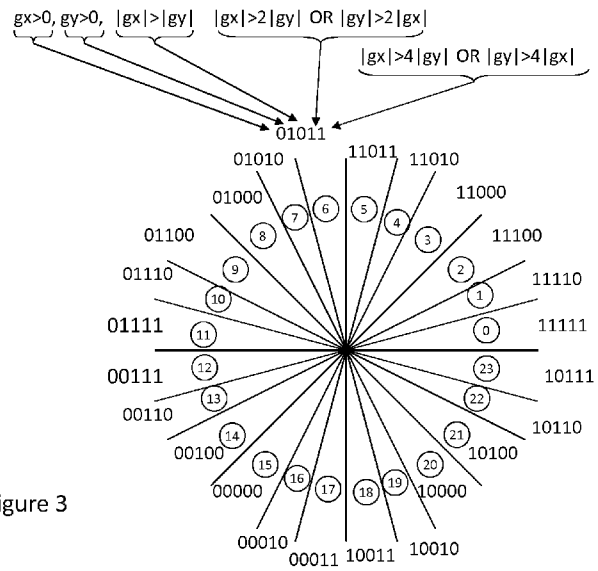


Figure 3

(57) Abstract: A method for producing a histogram of oriented gradients (HOG) for at least a portion of an image comprises dividing said image portion into cells, each cell comprising a plurality of image pixels. For each image pixel of a cell, a horizontal gradient component, gx, and a vertical gradient component, gy, is obtained based on differences in pixel values along at least a row of said image and a column of said image respectively including the pixel. A gradient is allocated to one of a plurality of sectors, each sector extending through a range of orientation angles. At least some of said sectors are divided from adjacent sectors along lines including $gx=2^n.gy$, where n is any integer value with a magnitude greater than or equal to 1. At least one sector is associated with a bin; and a count of each instance of a pixel gradient of a cell associated with a bin is performed to provide a HOG for said cell.

WO 2016/083002 A1

A METHOD FOR PRODUCING A HISTOGRAM OF ORIENTED GRADIENTS

Field of the Invention

The present invention relates to a method for producing a histogram of oriented gradients.

Background

Navneet Dalal and Bill Triggs, "Histograms of Oriented Gradients for Human Detection", IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol 1, page 886-893, 25-26 June 2005; and Navneet Dalal "Finding People in Images and Videos" PhD Thesis, L'institut National Polytechnique De Grenoble, 17 July 2006, disclose Histogram of Oriented Gradient (HOG) descriptors for use in image processing including for object detection and/or classification.

Figure 1 shows a simple example for calculating a HOG for a localized portion of an image, referred to herein as a cell.

In the example of Figure 1, a 2x2 pixel cell is employed. Each pixel value is combined with a 3x1 kernel to determine the respective horizontal components of the gradients for the pixels of the cell; and with a 1x3 kernel to determine the vertical components of the gradients for the pixels of the cell. (It will therefore be appreciated that pixel values from a border of one pixel around the 2x2 cell are needed in order to calculate the gradients for each pixel of the cell.) In the example of Figure 1, the pixels values for the corners of the border, 4, 0, 7 and 4 are shown for information only as these are not required for the exemplary kernels of Figure 1. However, it will be appreciated that other kernels could be employed which might require these or other pixels.

The horizontal and vertical gradient components are combined to provide vector gradients for each pixel of the cell. Using the gradients, a magnitude and orientation of each pixel gradient for the cell can be calculated. In the example of Figure 1, the gradients are mapped to one of 8 bins and a HOG can then be determined by counting occurrences of gradient orientation. In the example, 3 vectors are in bin 2 and 1 in bin 3 indicating the gradient for this portion of the

image is generally in the direction of bin 2. (This is not shown aligned with the values shown in the sample input image.) The magnitude values for each pixel gradient can then be used to determine the extent of the gradient in that direction.

It is this combination of HOG and magnitude values that is referred to herein as the HOG descriptors for a cell.

In Figure 1, the pixels values used comprise luminance values, for example, a Y component for a YCC format image. In other implementations, color gradients might also be calculated based on this or other image formats.

As will be seen from the example of Figure 1, simple applications based on HOG descriptors would require the use of:

- multipliers – to calculate the magnitude of the gradients; and
- arc tangent function to calculate the orientation of the gradients.

Clearly optimization of each of these calculations would allow for a more rational implementation for providing HOG descriptors and in particular this would allow for an efficient hardware implementation providing HOG descriptors in devices with limited processing capability such as hand-held cameras or smartphones.

Ryoji Kadota et al, “Hardware Architecture for HOG Feature Extraction”, Proceedings of 5th International Conference on Intelligent Information Hiding and Multimedia Signal Processing, 2009 pages 1330-1333, discloses several methods to simplify the computation of HOG descriptors, such as conversion of the division, square root and arctangent to simpler operations.

Tam Phuong Cao et al, “Implementation of real-time pedestrian detection on FPGA”, Image and Vision Computing New Zealand, 2008, 23rd International Conference, p1-6 discloses dividing gradient orientations into non-uniform angular bins.

Summary

According to the present invention there is provided a method for producing a histogram of oriented gradients according to claim 1.

In a further aspect there is provided an image processing system including logic for implementing the method of claim 1.

In a still further aspect, there is provided an image processing system as claimed in claim 26.

Embodiments of the present invention are suitable for efficient provision in hardware of HOG descriptors. The invention does not require multipliers and arc tangent function for implementation. It only requires adders and comparators, resulting in a fast hardware implementation with a small footprint.

Embodiments of the invention also provide a very flexible way of combining HOG sectors into bins.

Brief Description of the Drawings

Various embodiments of the present invention will now be described, by way of example, with reference to the accompanying drawings, in which:

Figure 1 illustrates a conventional calculation of a HOG for a cell;

Figure 2 show a quadrant split into 6 unequal sectors according to an embodiment of the present invention;

Figure 3 illustrates a technique for determining in which of 24 sectors a gradient lies according to an embodiment of the present invention;

Figure 4 illustrates an arrangement of sectors split according to Figures 2 and 3 and combined into 8 cardinal and inter-cardinal bins;

Figure 5 illustrates one quadrant of another arrangement of sectors split according to Figures 2 and 3 and combined into 12 almost equal bins;

Figure 6 illustrates exemplary logic for configurably combining sectors into bins;

Figure 7 shows a high level HOG image processing (IP) pipeline according to an embodiment of the present invention;

Figure 8 illustrates an internal architecture of a hardware implementation of the HOG IP of Figure 7 with a block of four line buffers;

Figure 9 illustrates an internal architecture of a hardware implementation of the HOG IP of Figure 7 with a block of two line buffers; and

Figure 10 illustrates windowing for successively determining respective HOGs for cells in a swath of an image.

Description of the Preferred Embodiments

Theoretically, to find the orientation of a gradient for a pixel, the arc tangent function is used:

$$\text{orientation} = \arctan(gx/gy)$$

where gx is the horizontal component and gy the vertical component of the gradient calculated for example, using the kernels shown in relation to Figure 1.

However, as will be appreciated from Figure 1, the exact orientation angle for a gradient is not required; instead there is typically only a requirement to assign the gradient for a pixel to a HOG orientation bin covering a range of angles.

Embodiments of the present invention provide a simple way of calculating in which one of a number of sectors a gradient orientation fits. While individual sectors could be used as HOG bins, it can be useful to combine sectors into orientation bins as will be explained in more detail below.

Referring to Figure 2, in the example, each of the four Cartesian quadrants are split into 6 sectors – one of which is shown. The boundaries between the sectors indicated by the radiating lines of Figure 2 are defined by the following conditions:

- a) gx=0; b) gx=4*gy; c) gx=2*gy; d) gx=gy ; e) gx=2*gy; f) gx=4*gy; g) gy=0

As will be seen, the resulting sectors do not have the same angular size.

In order to determine in which sector a gradient lies, rather than calculating the angle for each gradient, a limited number of comparisons can be performed including:

1) gx>0;

2) gy>0;

3) |gx|>|gy|;

4) |gx|>2|gy| OR |gy|>2|gx|;

5) |gx|>4|gy| OR |gy|>4|gx|

These 5 tests can be combined into a mask to determine in which one of 24 sectors a given gradient lies. So if gx > 0 the first bit of the mask is '1'; if gy > 0 the second bit is '1' and so on.

Figure 3 shows the mask values for each sector 0-23.

So, for example, where gx>0, the gradient will lie in one of sectors 5-18; and if gy>0, such a gradient could only lie in one of sectors 5-0. If in this case, |gx|>|gy|, the possibilities are limited to sections 2-0; and so on until a specific sector is determined.

It will be seen from the above, that simple gate logic can be employed to clock through gx and gy values for a given pixel and to receive a gradient sector value for the pixel.

It will also be noted that as sectors are determined using powers of 2 of gx and gy, tests 4) and 5) listed above can be performed in hardware using bit shifted versions of gx and gy instead of requiring multiplication (or division).

It will be appreciated that a further level of granularity could be provided by adding a test:

|gx|>8|gy| OR |gy|>8|gx|, however it is not thought that this would significantly improve results for most applications.

Equally, if for example, test 5 not used, granularity would be reduced, as it would not be possible to discriminate between sectors 0,1; 4,5; 6,7; 10,11; 12,13; 16,17; 18, 19; and 22,23 in the example of Figure 3. However, this could provide acceptable results for some applications.

Thus, in addition to or as an alternative to the x, y axes, the boundaries between sectors are defined by lines corresponding to $gx=2^n.gy$. The situation where n=0 is the boundary employed in implementations such as classic Dalal et al referenced above whereas combinations of boundaries based on values of $|n|\geq 1$ provide for effective hardware implementation of binning as described in more detail below.

Implementations using the above approach can allow any combinations of sectors to form a bin, so that:

- a bin can correspond to any number of sectors from 0 to 23;
- the sectors in a bin do not have to be adjacent;
- not all sectors must be part of a bin; and
- different bins can contain the same sector.

Figures 4 and 5 show exemplary bin configurations. In Figure 4, sectors 4-7 have been combined to indicate a N(orth) gradient; sectors 8-9 have been combined to indicate a NW gradient; sectors 10-13 have been combined to indicate a W gradient and so on. Figure 5 indicates that sectors can be combined into bins indicating different semantics such as horizontal, vertical, small slope or steep slope.

Figure 6 shows a hardware implementation where the tests 1) to 5) are implemented along with a configurable combination of sectors into any one or more of up to 16 bins. (The comparison logic for the tests and shifting logic for tests 4) and 5) is not shown.)

Thus, a 5 input AND gate output for sect_0 will be positive if all five tests are positive (11111 in Figure 3); while the 5 input AND gate output for sect_1 will be positive if tests 1) to 4) are positive and test 5) negative (11110 in Figure 3) and so on.

Each bin is defined with a respective 24 bit sector selection register comprising a mask defining the one or more sectors which combine to provide a given bin. For the sector selection register for bin_0_sel, each sector selection register bit is ANDed with the output of a corresponding 5 input AND gate providing the output for a sector in a set of AND gates B0...B23 (only three of which are shown). The outputs of the AND gates B0...B23 are combined in a common OR gate so that if a gradient falls in any configured sector of bin_0_sel configured according to sector_sel_reg_0, the OR gate output will indicate that the gradient for the pixel falls in that bin. The gates B0...B23 are replicated for each bin from bin_0_sel to bin_15_sel and connect with the outputs of the 5 input AND gates and a respective sector selection register from sector_sel_reg_1 to sector_sel_reg_15.

Thus, for any given pixel, a 16 bit bin mask, bin_xx_sel, where xx=0 to 15, can be produced in the same clock cycle as gx and gy are presented to the logic.

At the same time the magnitude of the gradient could be simply calculated using Pythagoras theorem:

$$mag = \sqrt{gx^2+gy^2}$$

However, the exact magnitude value is typically not essential for HOG calculation and usage. So in order to rationalise the hardware implementation, multipliers (and divisors) can be avoided by using the following approximation for gradient magnitude for a pixel:

$$mag = |gx| + |gy| - \min(gx, gy)/2$$

Thus, again for any clocked values of gx and gy, the magnitude of the pixel can be produced in the same clock cycle.

Referring now to Figure 7, in one hardware implementation, a HOG image processing (IP) pipeline provides two main interfaces:

- Input image interface, receiving one image pixel per clock cycle in raster order; and

- Output interface providing HOG descriptors for up to one cell per clock cycle, depending on the configured step between cells.

A typical clock frequency for such a module would be about 400MHz.

Protocol wrappers can be provided for the input and output interface, to allow the HOG IP block to be connected to any kind of interface and system which might require HOG descriptors for an image or any rectangular portion of an image.

Equally, the HOG IP pipeline interface can enable features such as step size, cell size etc to be specified.

In general, the image processing pipeline splits an image or image portion into cells on the fly. Typically, an input image would be split into rectangular cells and HOG descriptors calculated for each cell.

The pipeline can work with any cell size. It has been found that 4x4 cells are a good choice and HOGs for such cells can be readily combined to provide HOGs for larger cells as will be explained later. While cell sizes less than 4 pixels wide or high might be regarded as too small to provide useful information, basing the hardware on a larger basic cell size than 4x4 would increase the gate count and the number of line buffers required. Where HOG descriptors provided for adjacent single cells are combined to provide HOG values for larger cells within an image, as described below, HOGs can be determined for larger cells, but without increasing gate count. It will also be appreciated that the steps from cell to cell can be configured. For example, a step of 1 on the horizontal and vertical directions means that a new HOG is produced at the HOG IP output interface every clock cycle (except when moving to a new row).

Hardware could also be arranged to support independent horizontal and vertical steps between cells. The hardware could in addition or alternatively be arranged so that HOG descriptors are provided for overlapping cells within an image (as would be the case for a 4x4 cell size with a step size of 1 or 2).

We would remark that a step size larger than the cell size is typically not useful. Also, a step size of 3 is typically not required and supporting such a step size would also lead to more complicated hardware implementations.

In any case, Figures 8 and 9 illustrate two exemplary hardware implementations of the HOG IP of Figure 7. Both implementation use the same sub-blocks of logic, but differ in the way line buffers are arranged. The implementation of Figure 8 requires less memory for line buffering, but needs more logic gates (25 KGates for Figure 8 as distinct from 20KGates for Figure 9). In Figure 8, all line buffers can be grouped into a contiguous region of RAM whereas in the implementation of Figure 9 two separate sets of line buffers are employed.

Turning to Figure 8, in this case, an input line buffer including pixels values from 5 previous rows LB0...LB5 is read sequentially into memory one pixel at a time. Recalling that a border pixel of 1 is required for the kernels shown in Figure 1, once the first 5 lines of data have been read into memory, as the 6th line is being read, HOG descriptors for a first swath of 4x4 cells of the image corresponding to rows 2-5 can be calculated using the simple [-1, 0, 1] horizontal and vertical differentiation kernels of Figure 1.

As each pixel P in the 6th row is read, the horizontal and vertical gradient components gx,gy can be provided for the corresponding pixels P1...P4 in the previous 4 rows of the image i.e. rows which are displaced by n*pixel_columns, where n=1...4, from the pixel P within the line buffer.

Using the circuitry of Figure 7 and the calculation for magnitude explained above, the (16 bit) bin mask and magnitude based on these values of gx,gy for pixels P1...P4 is immediately available and this is shifted into a 5x4 windowed buffer which maintains the values for 5 columns of the image.

As each pixel for a row is clocked in through the HOG IP input interface, a set of 4 bin masks and magnitudes is shifted into a column of the 5x4 windowed buffer with the previous values shifted out.

A Cell HOG calc block comprises a counter circuit which tallys the count for each bin for each pixel comprising a 4x4 set of cells read from the 5x4 windowed buffer. This can be done by simply adding the corresponding bits of each of the 16 bin_xx_sel registers generated using the circuitry of Figure 7.

Thus, the Cell HOG calc block produces an array of 16 counts for each of configured bin_0 to bin_15 for each clock cycle.

Referring to Figure 10, a particularly useful approach using the 5x4 windowed buffer is for the Cell HOG calc block to subtract values from the oldest column A of the 5x4 windowed buffer and add the values for the newest column B of the 5x4 windowed buffer to the previous values of a 4x4 set of cells read from the 5x4 windowed buffer to obtain the current HOG at each clock cycle. This allows the Cell HOG calc block to be hardwired to the windowed buffer and to provide a HOG on each clock cycle without shifting or switching.

Referring back to Figure 8, an array comprising the counts for each of the 16 bins comprising the HOG is provided to a decimation block on each clock cycle.

The decimation block reduces the frequency of HOGs produced, according to the horizontal and vertical steps. So for example with a horizontal step size of 4, the HOG IP output interface would only produce a new HOG or update the HOG every fourth clock cycle.

It will also be appreciated that for a horizontal and vertical step size of 4, once a given swath of an image is completed, 3 new lines of the image will need to be read, before on the 5^{th} pixel of next line, the next HOG becomes available.

In the case of Figure 9, the input line buffer only includes pixels values from 2 previous rows LB0 and LB1 and so when a pixel P is being read, only the horizontal and vertical gradient components for a corresponding pixel P1 from the previous line can be determined.

In this case, each newly calculated value of magnitude and bin mask for the pixel P1 is shifted into a 5x4 windowed buffer one at a time (rather than 4 at a time in Figure 8). As the values of magnitude and bin mask for the previous 3 lines of the image are shifted out of this buffer, they

are stored in a second set of Magnitude and Bin Line buffers MBL0…MBL2, so that as the process advances row by row through an image, the magnitude and bin mask values from the previous 3 lines are always available for calculating a HOG for a 4x4 cell. (The memory required for each Magnitude and Bin Line buffer MBL0…MBL2 is greater than for the line buffers LB0…LB1 (and indeed LB2…L4) because for each pixel both the gradient magnitude and bin mask need to be stored.) Nonetheless, for a given set of values in the 5x4 windowed buffer, the process and circuitry of Figure 9 is as per Figure 8.

For simplicity, neither Figure 8 or 9 illustrate the processing of the magnitude values available in the 5x4 windowed buffer – these can be read as required through the HOG IP output interface when the HOG values for a cell are being read to provide a complete set of HOG descriptors for a cell.

It will be appreciated that using the above described implementations the HOG values for corresponding bins of adjacent 4x4 cells can simply be added to provide HOG values for larger cells within an image. Thus, with one pass over an image, an application can perform top down or bottom up analysis of gradient as required.

Many variants of the above described embodiments are possible.

For example, the magnitude values for a bin corresponding to all 24 sectors could be used to give a general idea about how steep the gradients in the cell are and can be used for scaling for example.

Claims

1. A method for producing a histogram of oriented gradients (HOG) for at least a portion of an image comprising:

dividing said image portion into cells, each cell comprising a plurality of image pixels;

for each image pixel of a cell,

obtaining a horizontal gradient component, gx, and a vertical gradient component, gy, based on differences in pixel values along at least a row of said image and a column of said image respectively including said pixel; and

allocating a gradient to one of a plurality of sectors, each sector extending through a range of orientation angles and at least some of said sectors being divided from adjacent sectors along lines including $gx=2^n.gy$, where n is any integer value with a magnitude greater than or equal to 1;

associating at least one sector with a bin; and

counting each instance of a pixel gradient of a cell associated with a bin to provide a HOG for said cell.

2. A method according to claim 1 further comprising, for each image pixel of a cell, calculating a magnitude of said gradient.

3. A method according to claim 2 wherein said calculating said magnitude is based on the formula:

$$mag = |gx| + |gy| - \min(gx, gy)/2.$$

4. A method according to claim 1 wherein said sectors are divided from adjacent sectors along the following lines: gx=0; gx=4*gy; gx=2*gy; gx=gy ; gx=2*gy; gx=4*gy; and gy=0.

5. A method according to claim 1 wherein each cell comprises 4x4 image pixels.

6. A method according to claim 1 wherein each cell is displaced by steps of one of 1, 2 or 4 pixels vertically or horizontally from an adjacent cell.

7. A method according to claim 1 wherein said obtaining a gradient component comprises subtracting a pixel value for a first pixel immediately adjacent a pixel from a pixel value for a second pixel immediately adjacent said pixel and opposite first said pixel.

8. A method according to claim 1 comprising associating at least one bin with a plurality of sectors.

9. A method according to claim 8 wherein said sectors are either adjacent or non-adjacent.

10. A method according to claim 8 wherein a plurality of bins are associated with a given sector.

11. A method according to claim 8 wherein not all sectors are associated with a bin.

12. A method according to claim 1 comprising associating each sector with a respective bin.

13. A method according to claim 1 wherein said pixel values are intensity values.

14. A method according to claim 1 wherein said pixel values are colour values.

15. A method according to claim 1 further comprising the step of combining respective HOGs for a plurality of adjacent cells.

16. An image processing system including digital logic for implementing the method of claim 1.

17. An image processing system according to claim 16 wherein said logic includes a plurality of line buffers and wherein said logic is arranged to receive an image pixel-by-pixel on each cycle of a clock and to accumulate previous pixel values in said line buffers.

18. An image processing system according to claim 17 wherein said logic is arranged to produce said horizontal and vertical gradient components for one or more pixels stored in said line buffers in the same column of an image as a currently received pixel.

19. An image processing system according to claim 18 wherein said logic includes a buffer arranged to accumulate indicators for which bins each pixel of a cell is associated with.

20. An image processing system according to claim 19 wherein said logic comprises a counter connected to said buffer for providing said count of pixel gradients of a cell associated with respective bins.

21. An image processing system according to claim 20 wherein said logic is arranged to provide said count synchronously in multiple steps of said clock cycle.

22. An image processing system according to claim 16 including logic for combining tests for whether a gradient for a pixel lies within the boundaries of any of said sectors.

23. An image processing system according to claim 22 wherein said tests comprise:

gx>0; gy>0; |gx|>|gy|; |gx|>2|gy| OR |gy|>2|gx|; and |gx|>4|gy| OR |gy|>4|gx|.

24. An image processing system according to claim 22 wherein said logic for combining tests is replicated for each sector.

25. An image processing system according to claim 22 including logic for combining said test results with respective bin configurations, each bin configuration associating a bin with a combination of said sectors, to provide a bin mask indicating any configured bins in which said gradient lies.

26. An image processing system including digital logic for producing a histogram of oriented gradients (HOG) for respective cells from at least a portion of an image, each cell comprising a plurality of image pixels:

said logic including a plurality of line buffers and being arranged to receive an image pixel-by-pixel on each cycle of a clock and to accumulate previous pixel values in said line buffers,

said logic being arranged to produce a horizontal gradient component, gx, and a vertical gradient component, gy, for one or more pixels stored in said line buffers from the same column of an image as a currently received pixel based on differences in pixel values along at least a row of said image and a column of said image respectively including said one or more pixels,

said logic being arranged to allocate respective gradients for said one or more pixels to one of a plurality of bins, each bin extending through a range of orientation angles,

said logic including a buffer arranged to accumulate indicators for which bins each pixel of a cell is associated with, and

said logic including a counter connected to said buffer for providing a count of pixel gradients of a cell associated with respective bins, said logic being arranged to provide said count synchronously in multiple steps of said clock cycle according to cell displacement.

27. An image processing system according to claim 26 wherein said logic for allocating respective gradients includes logic for testing if:

$gx>0$; $gy>0$; $|gx|>|gy|$; $|gx|>2|gy|$ OR $|gy|>2|gx|$; and $|gx|>4|gy|$ OR $|gy|>4|gx|$.

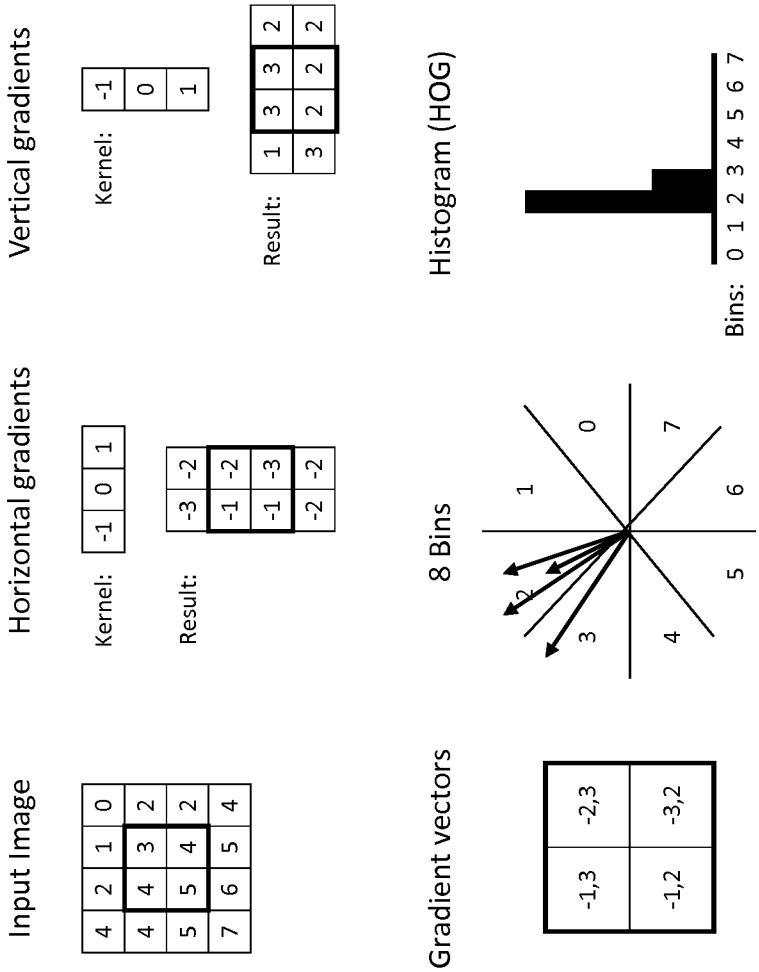


Figure 1 (Prior Art)

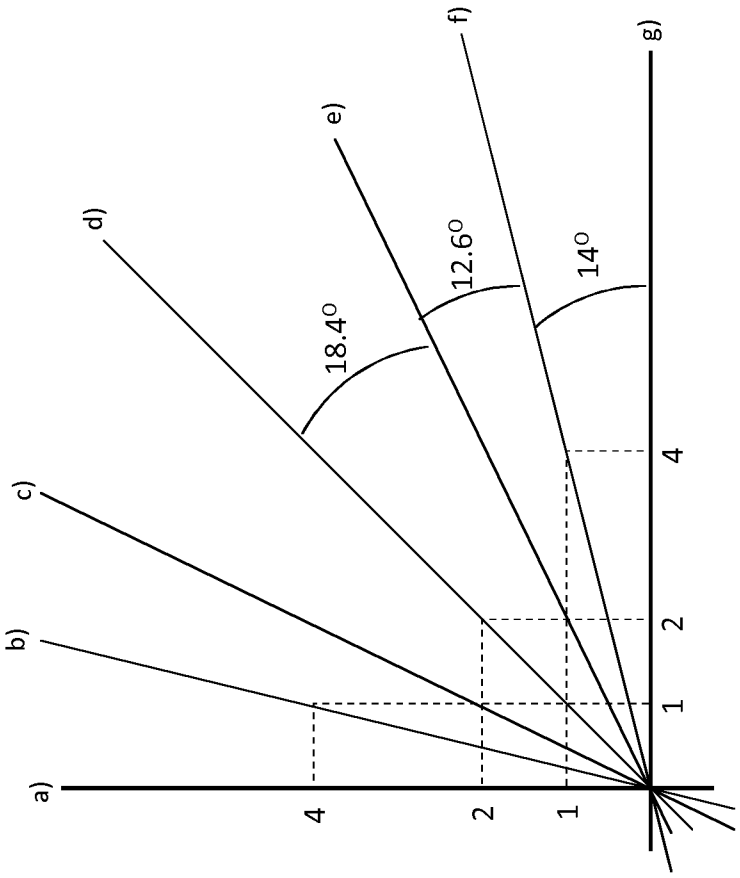


Figure 2

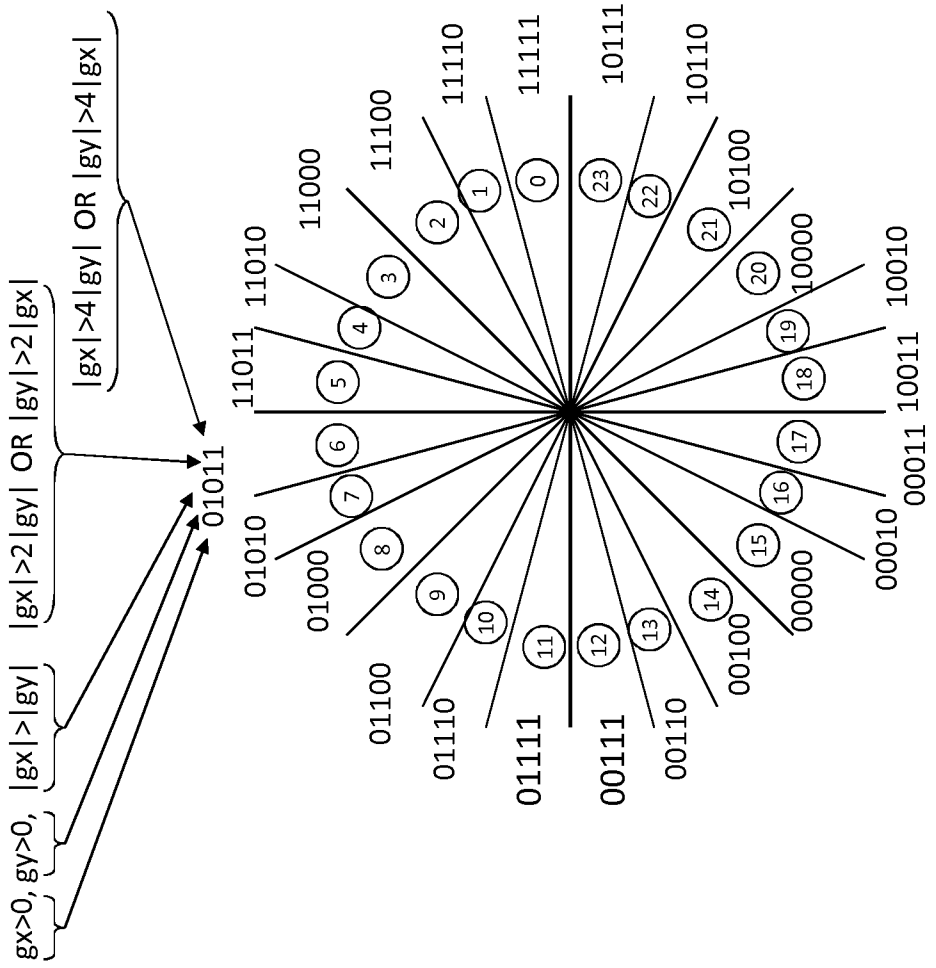

gx>0, gy>0, |gx|>|gy| |gx|>2|gy| OR |gy|>2|gx|
|gx|>4|gy| OR |gy|>4|gx|
01011
11011
11010
11000
11100
11110
11111
10111
10110
10100
10000
10010
10011
00011
00010
00000
00100
00110
00111
01111
01110
01100
01000
01010
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23

Figure 3

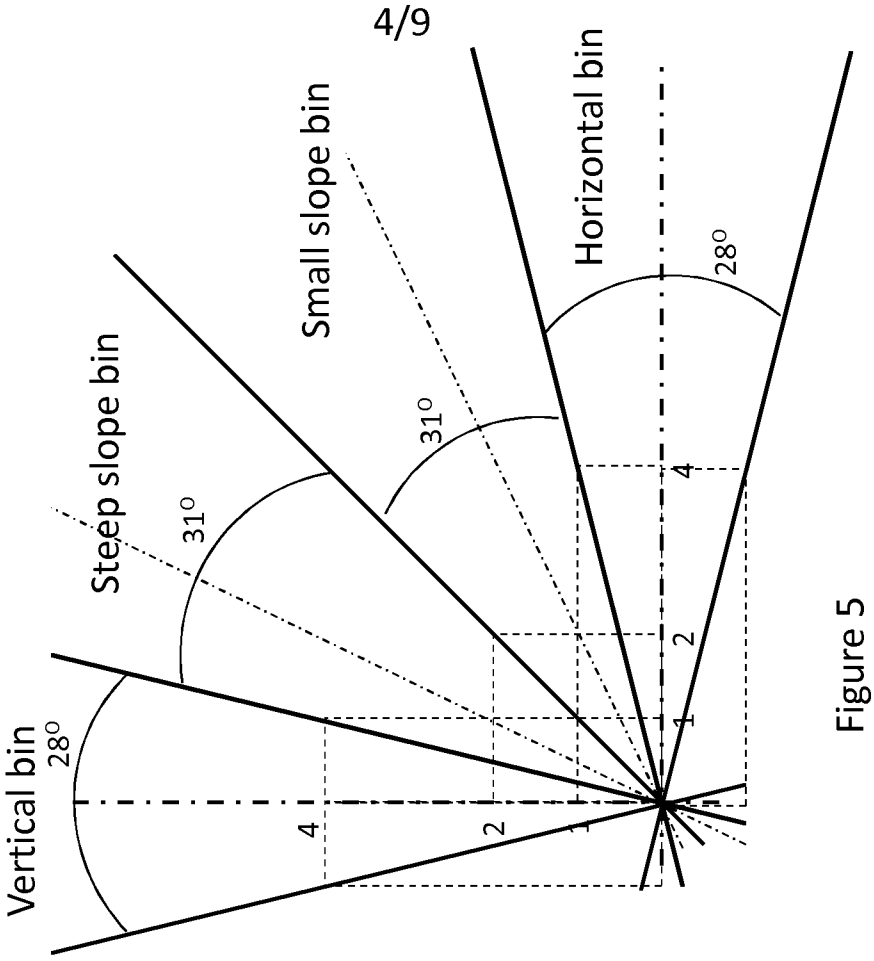


Figure 5

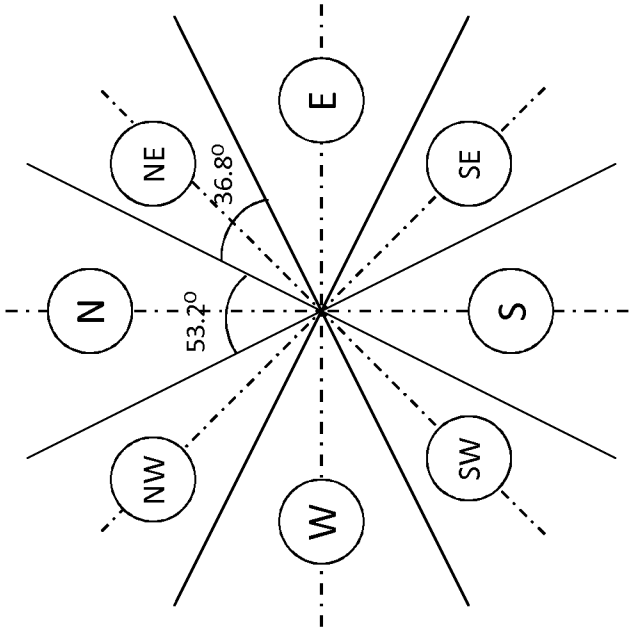
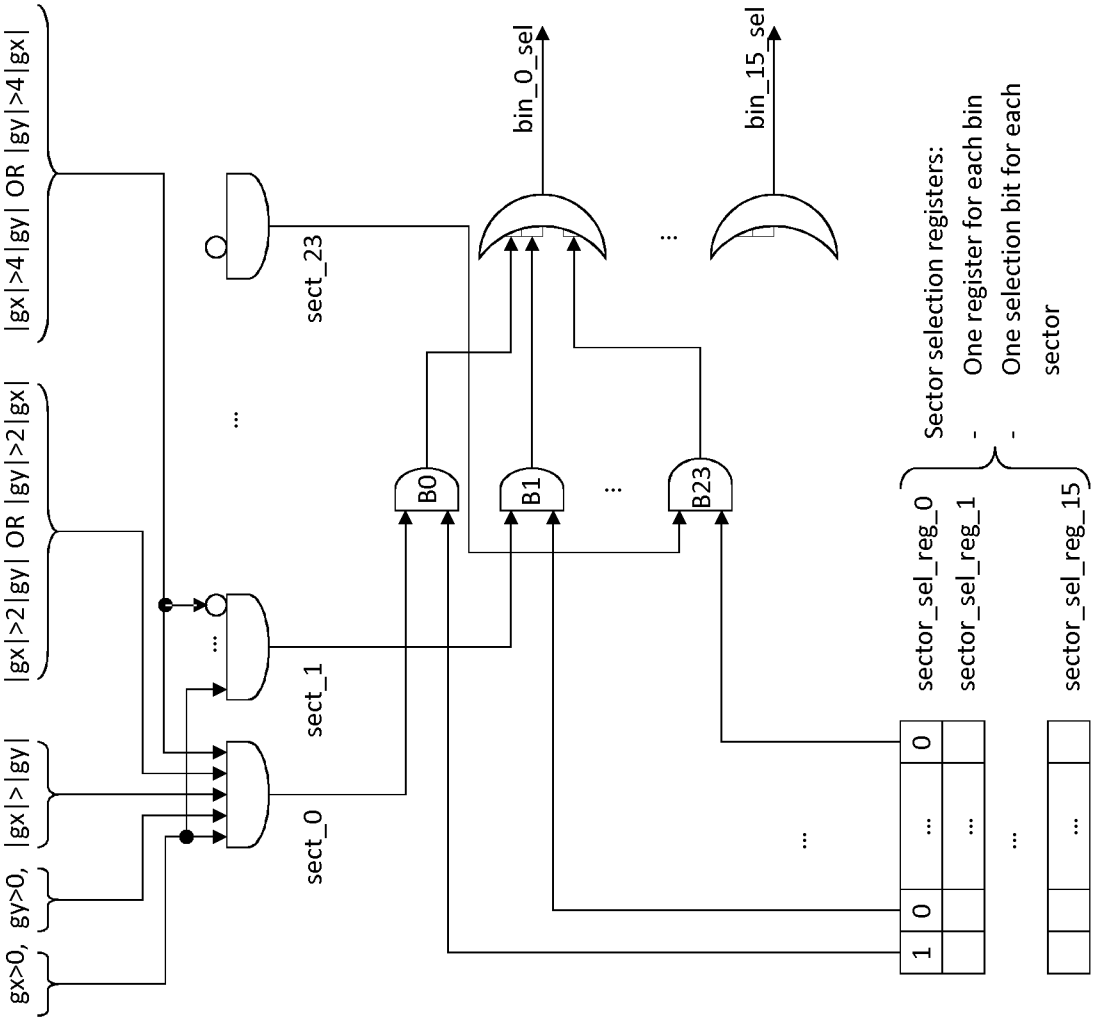


Figure 4

Figure 6


gx>0, gy>0, |gx|>|gy|
|gx|>2|gy| OR |gy|>2|gx|
|gx|>4|gy| OR |gy|>4|gx|
sect_0
sect_1
sect_23
B0
B1
B23
bin_0_sel
bin_15_sel
Sector selection registers:
- One register for each bin
- One selection bit for each sector
sector_sel_reg_0
sector_sel_reg_1
sector_sel_reg_15
1
0
0

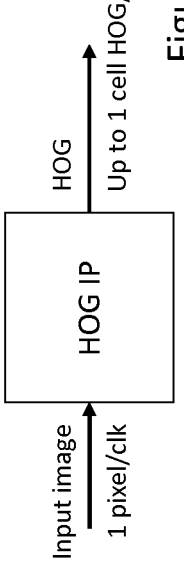


Figure 7

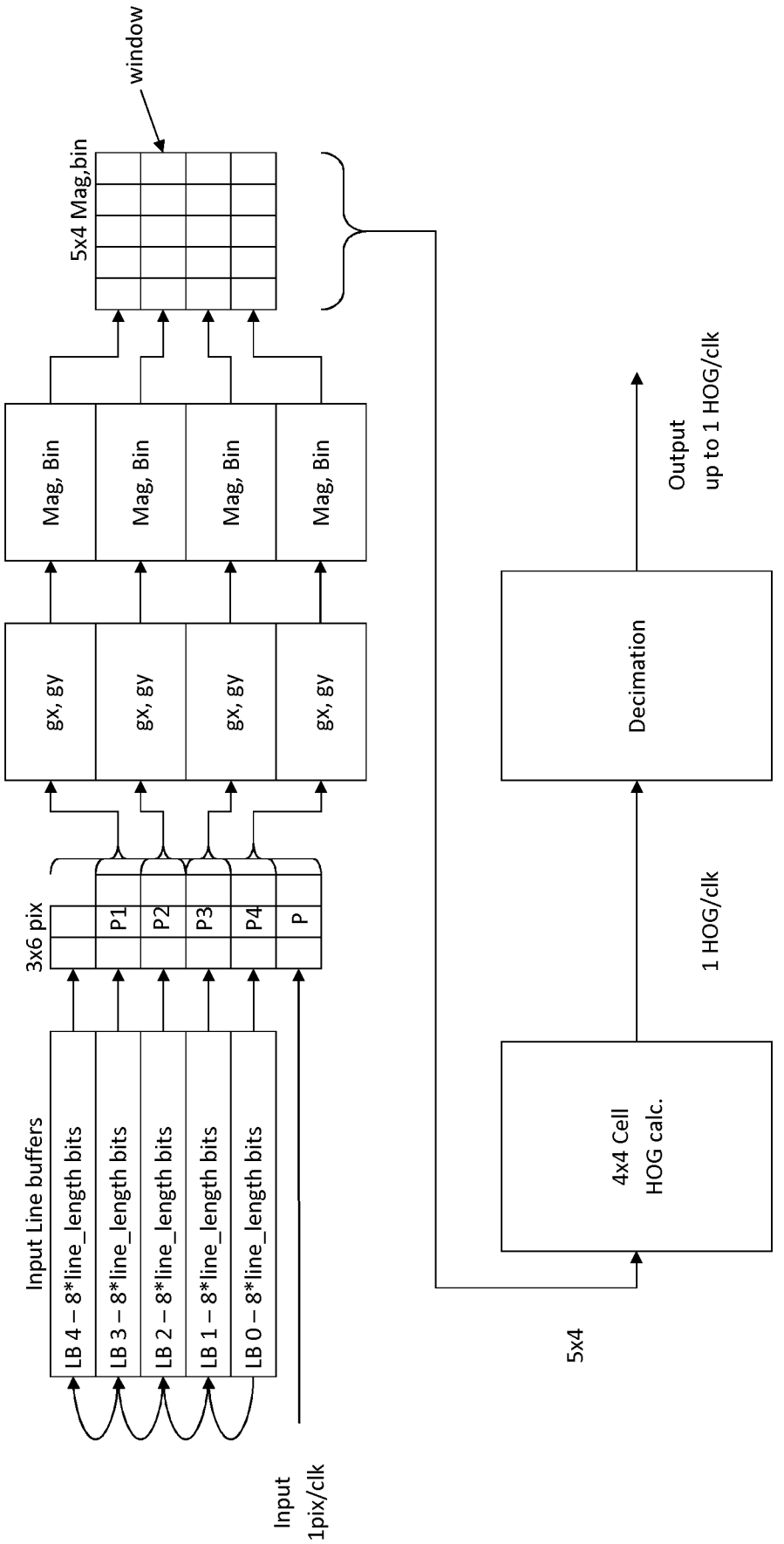
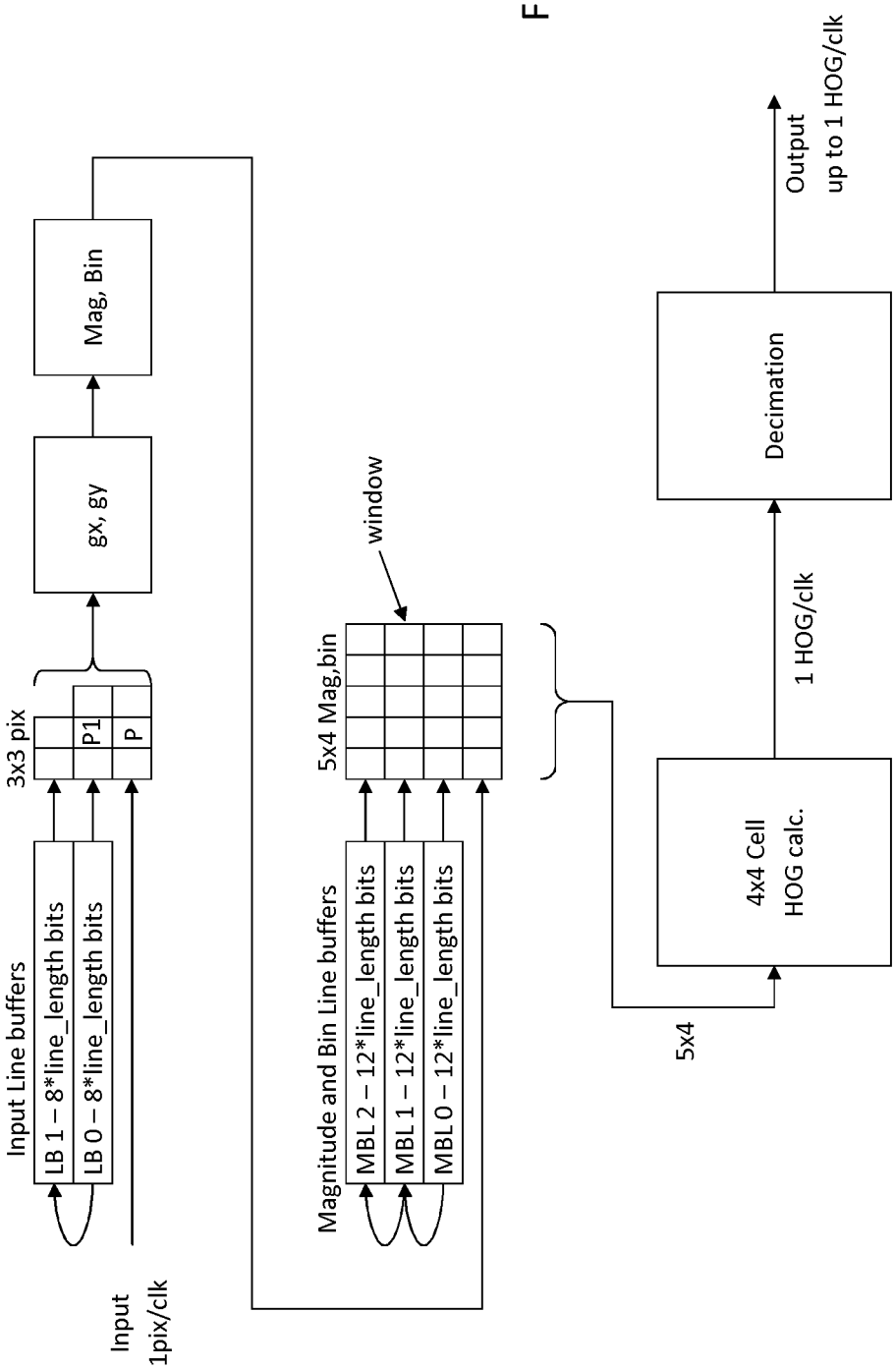

window
5x4 Mag,bin
Mag, Bin
Mag, Bin
Mag, Bin
Mag, Bin
gx, gy
gx, gy
gx, gy
gx, gy
3x6 pix
P1
P2
P3
P4
P
Input Line buffers
LB 4 – 8*line_length bits
LB 3 – 8*line_length bits
LB 2 – 8*line_length bits
LB 1 – 8*line_length bits
LB 0 – 8*line_length bits
Input
1pix/clk
5x4
4x4 Cell
HOG calc.
1 HOG/clk
Decimation
Output
up to 1 HOG/clk

Figure 8

Figure 9


Input
1pix/clk
Input Line buffers
LB 1 – 8*line_length bits
LB 0 – 8*line_length bits
3x3 pix
P1
P
gx, gy
Mag, Bin
Magnitude and Bin Line buffers
MBL 2 – 12*line_length bits
MBL 1 – 12*line_length bits
MBL 0 – 12*line_length bits
5x4 Mag,bin
window
5x4
4x4 Cell
HOG calc.
1 HOG/clk
Decimation
Output
up to 1 HOG/clk

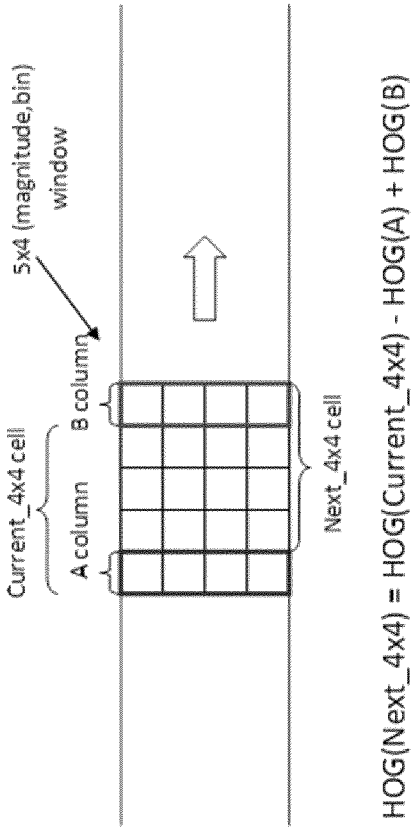

5x4 (magnitude,bin)
window
B column
Current_4x4 cell
A column
Next_4x4 cell
HOG(Next_4x4) = HOG(Current_4x4) - HOG(A) + HOG(B)

Figure 10

INTERNATIONAL SEARCH REPORT

International application No.
PCT/EP2015/073058

Box No. II Observations where certain claims were found unsearchable (Continuation of item 2 of first sheet)

This international search report has not been established in respect of certain claims under Article 17(2)(a) for the following reasons:

1. ☐ Claims Nos.:
because they relate to subject matter not required to be searched by this Authority, namely:

2. ☐ Claims Nos.:
because they relate to parts of the international application that do not comply with the prescribed requirements to such an extent that no meaningful international search can be carried out, specifically:

3. ☐ Claims Nos.:
because they are dependent claims and are not drafted in accordance with the second and third sentences of Rule 6.4(a).

Box No. III Observations where unity of invention is lacking (Continuation of item 3 of first sheet)

This International Searching Authority found multiple inventions in this international application, as follows:

see additional sheet

1. ☐ As all required additional search fees were timely paid by the applicant, this international search report covers all searchable claims.

2. ☐ As all searchable claims could be searched without effort justifying an additional fees, this Authority did not invite payment of additional fees.

3. ☐ As only some of the required additional search fees were timely paid by the applicant, this international search report covers only those claims for which fees were paid, specifically claims Nos.:

4. ☒ No required additional search fees were timely paid by the applicant. Consequently, this international search report is restricted to the invention first mentioned in the claims; it is covered by claims Nos.:

1-16, 22, 23, 27

Remark on Protest

☐ The additional search fees were accompanied by the applicant's protest and, where applicable, the payment of a protest fee.

☐ The additional search fees were accompanied by the applicant's protest but the applicable protest fee was not paid within the time limit specified in the invitation.

☐ No protest accompanied the payment of additional search fees.

INTERNATIONAL SEARCH REPORT

International application No
PCT/EP2015/073058

A. CLASSIFICATION OF SUBJECT MATTER
INV. G06K9/00 G06K9/46
ADD.

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
G06K

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)
EPO-Internal

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	CHEN PEI-YIN ET AL: "An Efficient Hardware Implementation of HOG Feature Extraction for Human Detection", IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS, IEEE, PISCATAWAY, NJ, USA, vol. 15, no. 2, 1 April 2014 (2014-04-01), pages 656-662, XP011544599, ISSN: 1524-9050, DOI: 10.1109/TITS.2013.2284666 [retrieved on 2014-04-03] the whole document ----- -/--	1-16,22, 23,27

[X] Further documents are listed in the continuation of Box C. [ ] See patent family annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
15 January 2016	07/03/2016

Name and mailing address of the ISA/	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Fax: (+31-70) 340-3016	Neubüser, Bernhard

2

Form PCT/ISA/210 (second sheet) (April 2005)

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	NIKLAS PETTERSSON ET AL: "The histogram feature - a resource-efficient Weak Classifier", INTELLIGENT VEHICLES SYMPOSIUM, 2008 IEEE, IEEE, PISCATAWAY, NJ, USA, 4 June 2008 (2008-06-04), pages 678-683, XP031318819, ISBN: 978-1-4244-2568-6 section III: Implementation	1-16,22, 23,27
X	TAM PHUONG CAO ET AL: "Real-Time Vision-Based Stop Sign Detection System on FPGA", COMPUTING: TECHNIQUES AND APPLICATIONS, 2008. DICTA '08.DIGITAL IMAGE, IEEE, PISCATAWAY, NJ, USA, 1 December 2008 (2008-12-01), pages 465-471, XP031371914, ISBN: 978-0-7695-3456-5 page 467	1-16,22, 23,27
A	DALAL N ET AL: "Histograms of Oriented Gradients for Human Detection", PROCEEDINGS / 2005 IEEE COMPUTER SOCIETY CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION, CVPR 2005 : [20 - 25 JUNE 2005, SAN DIEGO, CA], IEEE, PISCATAWAY, NJ, USA, vol. 1, 20 June 2005 (2005-06-20), pages 886-893, XP010817365, DOI: 10.1109/CVPR.2005.177 ISBN: 978-0-7695-2372-9 cited in the application the whole document	1-16,22, 23,27
X	Sebastian Bauer ET AL: "FPGA Implementation of a HOG-based Pedestrian Recognition System FPGA Implementation of a HOG-based Pedestrian Recognition System", MPC Workshop, 1 July 2009 (2009-07-01), pages 1-10, XP055241059, Retrieved from the Internet: URL:http://www5.informatik.uni-erlangen.de/Forschung/Publikationen/2009/Bauer09-FIO.pdf [retrieved on 2016-01-13] section 4: "Implementation"	1-16,22, 23,27

-/--

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	John-Olof Nilsson: "Efficient implementation of data binning in sectors", Technical Report: TRITA-EE 2012:041, 1 January 2012 (2012-01-01), pages 1-33, XP055240925, Stockholm Retrieved from the Internet: URL:http://www.diva-portal.org/smash/get/diva2:548677/FULLTEXT01.pdf [retrieved on 2016-01-13] the whole document -----	1-16,22, 23,27
X	KAZUHIRO NEGI ET AL: "Deep pipelined one-chip FPGA implementation of a real-time image-based human detection algorithm", FIELD-PROGRAMMABLE TECHNOLOGY (FPT), 2011 INTERNATIONAL CONFERENCE ON, IEEE, 12 December 2011 (2011-12-12), pages 1-8, XP032096830, DOI: 10.1109/FPT.2011.6132679 ISBN: 978-1-4577-1741-3 page 3 - page 4 -----	1-16,22, 23,27

2

FURTHER INFORMATION CONTINUED FROM PCT/ISA/ 210

This International Searching Authority found multiple (groups of) inventions in this international application, as follows:

1. claims: 1-16, 22, 23, 27

A method for producing a histogram of oriented gradients, HOG, for at least a portion of an image as known from Dalal and Triggs, D??, wherein at least one sector, corresponding to a bin of gradient directions, is bounded by a line of points (x,y) satisfying x = m * y, wherein m is an integer power of 2, and m is different from 1.

1.1. claims: 16, 22

A general purpose computer.

---

2. claims: 17-21, 26

An image processing system including digital logic including a plurality of line buffers, wherein the logic is arranged to receive an image pixel-by-pixel on each cycle of a clock and to accumulate previous pixel values in said line buffers.

---

3. claim: 24

An image processing system comprising a parallel processing sub-architecture for combining the results of certain tests.

---

4. claim: 25

An image processing system comprising a configurable logic combiner which combines the results of certain tests according to a given configuration.
