A High-Definition LIDAR System Based on Two-Mirror Deflection Scanners

Xuan Truong Nguyen, Van Luan Dinh, Hyuk-Jae Lee, and Hyun Kim, *Member, IEEE*

***Abstract*— This paper addresses the problem of adopting a state-of-art laser marking system with a two-mirror deflection scanner to make a high-definition light detection and ranging (LIDAR) system. To this end, a galvanometer scanner is modeled with parameterization and then the well-known raster scanning strategy is analyzed considering the physical scanning movement and the minimum spanning tree. As a result of this analysis, the relationship between the field of view (FOV) of the captured image and the scanning speed is clearly described. Furthermore, sufficient conditions are derived for an acquired image to fully cover the FOV and also for captured objects to be well aligned for a target frame rate. Finally, a prototype LIDAR system is developed to verify the proposed concepts and to prove that it successfully generates images at various resolutions depending on the target frame rates. Experimental results show that the scanner achieves the frame rates of 17.6, 9.0, and 4.6 frames per second fps for image sizes of 240 × 16, 240 × 32, and 240 × 64 resolutions, respectively.**

***Index Terms*— Laser scanner, galvanometer scanners, optimal scanning patterns, 3D lidar systems.**

I. INTRODUCTION

ACTIVE sensor technologies such as light detection and ranging (LIDAR)) has been intensively studied in theory and widely adopted in many applications, i.e., self-driving cars [1]–[4], robotics and smart sensors [5]–[7]. Furthermore, massive datasets such as KITTI [8], [9] also consist of ground-truth data which are generated from data sourced from a LIDAR system. To measure distances, a LIDAR system estimates the time interval between the emission of the light from the LIDAR and the arrival of the light reflected from a distant object. According to the number of emitter and detector pairs, LIDAR systems are categorized into two types: those which use one pair and those which use multiple pairs. The well-known commercial product Velodyne LIDAR [10] is motivated by a simultaneous use of multiple emitter/detector channels in the vertical direction. As those channels are rotated by a motor, the system can capture an image with multiple lines of data. For example, 16-, 32- or 64-channel versions can generate 16, 32 or 64 lines of point cloud data, respectively, corresponding to 300,000, 700,000 or 1,200,000 measurements per second. Moreover, they achieve a relatively high frame rate of about 10 frames per second (fps). Despite these advantages, their prices are relatively high due to the cost of the hardware for multiple channels. On the other hand, a single-channel LIDAR is widely used in laser marking systems [11], [12] or robotic applications [13]. Because there is only a single emitter and detector pair, the price of this system is relatively low. Moreover, the system is easy to control and flexible to scan an arbitrarily point in a field of view (FOV). As a result, it can be used to scan various patterns and to increase the vertical resolution. However, the scanning speed and frame rate are not very high, and consequently, it takes a few seconds to obtain patterns in a laser marker.

Manuscript received September 7, 2017; revised November 7, 2017; accepted November 8, 2017. Date of publication November 24, 2017; date of current version December 21, 2017. This work was supported in part by the Institute for Information and communications Technology Promotion grant through the Korea Government (MSIT) under Grant 2017-0-00721-001 (Development of intelligent semiconductor technology for vision recognition signal processing for vehicle based on multi-sensor fusion) and in part by the Research and Development Program of MOTIE/KEIT under Grant 10077609 (Developing Processor-Memory-Storage Integrated Architecture for Low Power, High Performance Big Data Servers). The associate editor coordinating the review of this paper and approving it for publication was Dr. Ioannis Raptis. *(Corresponding author: Hyun Kim.)*

The authors are with the Inter-university Semiconductor Research Center, Department of Electrical and Computer Engineering, Seoul National University, Seoul 08826, South Korea (e-mail: truongnx@capp.snu.ac.kr; luandv@capp.snu.ac.kr; hyuk_jae_lee@capp.snu.ac.kr; snusbkh0@capp.snu.ac.kr).

Digital Object Identifier 10.1109/JSEN.2017.2777500

A. Scope and Contributions

This paper addresses the problem of adopting a state-of-the-art laser marking system [14], [15], especially a two-mirror deflection scanner, to create a high-definition LIDAR system. The goal of this paper is to model galvanometer scanners and to analyze the performance of a LIDAR system based on this type of a scanner.

Three major contributions of this paper are described below.

1) Modeling: Section II presents the controller modeling of a galvanometer scanner in a LIDAR system by deriving the speed constraints affecting the scanner controller. Both timing constraints on the physical speed of the scanner and the latency of the communication interface are addressed. Furthermore, notations and common terminologies are introduced for the rest of the paper.

2) Scanning Problems: Section III defines the laser scanning problem, shows an intuitive solution, and proves its optimality. Hence, the relationship between the FOV of a captured image and the frame rate is derived. Moreover, sufficient conditions are derived to check whether the obtained image fully covers the FOV and includes well-aligned objects for given frame rates. Experimental results show that the scanners can achieve frame rates of 17.6, 9.0 and 4.6 fps for image sizes of 240×16, 240×32 and 240×64, respectively.

3) A LIDAR System: Section IV presents a sample LIDAR system which was developed based on the proposed solutions. The proposed system, able to provide images at various

TABLE I
LIST OF SOME LIDAR PRODUCTS ON THE MARKET

Model	Type	Numbers of channels	Scan frequency (Hz)	Horizontal Angle	Vertical Angle	Numbers of measurements
IBEO ALASCA [16]	3D	4	8-40	160^o	3.2^o	N/A
LEDDAR 16M [17]	2D	16	50	9^o-95^o	×	800
PUCK^{TM} VLP-16 [18]	3D	16	10	360^0	+15^o to -15^o	300,000
HDL-32E [19]	3D	32	10	360^0	+10^o to -30^o	700,000
HDL-64E [20]	3D	64	10	360^0	26.9^o	2,200,000
Riegl-VUX-1UAV [21]	2D	1	10-200	330^o	×	550,000

resolutions depending on target frame rates, is evaluated in terms of its speed and the resulting visual quality. The system achieves nearly 97,000 measurements per second while only using a single emitter/detector channel. In addition, given an FOV, scalable frame rates (i.e., 600 × 600, 300 × 300, or 150 × 150) can be achieved by the proposed system.

Finally, Section V concludes the paper.

B. Related Works

The demand for LIDAR sensors has been increasing due to the growing number of autonomous vehicles [7], [8]. These sensors play a critical role in self-driving cars by Google or Tesla. This subsection briefly reviews existing LIDAR sensors which are available as commercial products. To this end, exiting LIDAR products are compared with important performance parameters such as the number of channels, the scan frequency, and the horizontal/vertical angles. Table I summarizes the product specifications as given by the relevant companies. LEDDAR 16M [17] uses sixteen segments simultaneously to measure the distances of objects at sixteen angles. As its rate is 50Hz, the number of measurements per second is 800. Riegl-VUX-1UAV Lidar [21] uses a single pair of an emitter and a detector which can be rotated in a 330 ° FOV. This 2D LIDAR is able to scan 550,000 measurements per second, and the motor speed can be configured at different speeds between 10Hz and 200Hz. IBEO ALASCA [16] also adopts a similar rotating method, but it uses four channels. Therefore, it can scan a vertical angle of 3.2°. Velodyne LIDAR sensors [18]–[20] also use a rotation module to extend the vertical scanning angle. They increase the number of measurements proportionally by increasing the number of emitter/detector channels.

These LIDAR products share a similar scanning structure in which only one motor is used. Thus, any increase in the vertical resolution must depend on the number of optical channels. This paper suggests a solution which adopts the scanners used in laser marking systems. This LIDAR system only requires a single emitter/detector pair while offering various resolutions. To show the effectiveness of this LIDAR system, this paper aims to investigate its performance in terms of the frame rate, resolution and FOV and also creates a sample LIDAR system based on the proposed concept.

II. MODELLING A CONTROLLER OF GALVANOMETER SCANNERS

The common structure of a single-channel LIDAR system is a combination of 1) a scanning module and 2) a single-point measurement module. This paper focuses on the scanning module, which is directly related to the FOV and the frame rate of the system. Therefore, optical parts and depth measurements are only briefly covered in Section IV.

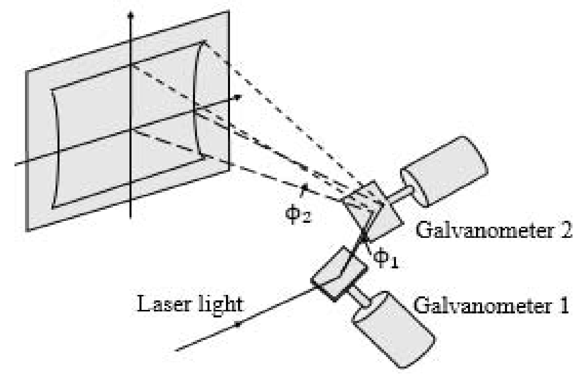


Fig. 1. Scanning FOV in a two-mirror deflection system.

A. Two-Mirror Deflection Scanners

The single-channel LIDAR system in this paper adopts the state-of-the-art laser marking system, i.e., the two-mirror deflection system of the type widely used in industrial applications [14], [15]. As shown in Fig. 1, the scanners control two motors to move their corresponding mirrors to a point at a specific position in the FOV. Each mirror is responsible for scanning the direction in the FOV. In Fig. 1, scanning angles which represents the angle differences with respect to the origins are defined by ϕ_1 and ϕ_2 for the x and y axes, respectively. In practice, it is known that two-mirror deflection systems are very accurate. For example, the scanners adopted in this paper have an angle step of 12 μrad, with each capable of scanning a maximum of 60,000 steps. Therefore, the system can cover a FOV of $41.2° \times 41.2°$ and represent a scene with very high-resolution images of $60,000 \times 60,000$.

B. Modelling the Controller of a Galvanometer Scanner

1) FOV Representation: Initially, let x_{min}, y_{min}, x_{max}, and y_{max} denote the minimum and maximum ranges in the vertical and horizontal directions of the interest FOV, respectively. In this case, these variables are expressed as follows:

$$0 \leq x_{min}, \quad x_{max}, \ y_{min}, \ y_{max} \leq 60,000 \tag{1}$$

In practice, the entire FOV is usually represented by a set of regular grid points. In other words, any two consecutive points

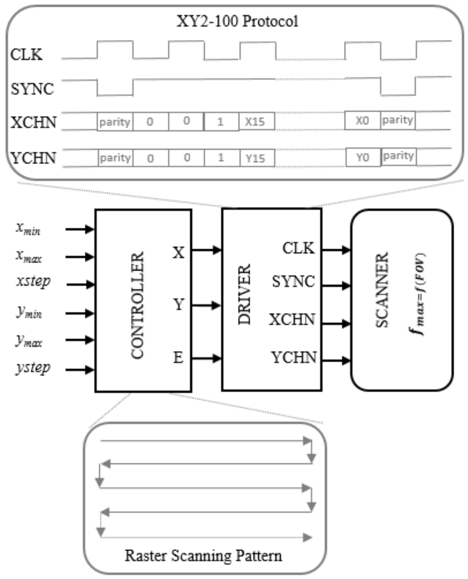


Fig. 2. Controller design for galvanometer scanners using the XY2-100 protocol and a raster scanning pattern.

are aligned by a fixed step. Let $xstep,\ ystep$ denote regular steps on the x-axis and y-axis of the FOV, respectively. Hence, the width and height of the obtained image can be derived as follows:

$$width = \frac{x_{max} - x_{min}}{xstep} \tag{2}$$

$$height = \frac{y_{max} - y_{min}}{ystep} \tag{3}$$

Example 1: It is assumed that the LIDAR system fully covers the FOV (*i.e.,* x_{min}=y_{min} $= 0$ and $x_{max} = y_{max} = 60,000$), which is represented by a frame 480×240 in size (*i.e.,* width $= 480$ and height $= 240$). The regular scanning steps are then as follows:

$$xstep = \frac{60,000}{480} = 125 \tag{4}$$

$$ystep = \frac{60,000}{240} = 250 \tag{5}$$

Similarly, an HD (1280×720) image can be achieved by setting $xstep = 46$ and $ystep = 83$. In particular, a specific frame size up to $60,000 \times 60,000$ can be obtained by our LiDAR system (see the detailed subjective images in Section IV-B).

2) Timing Constraints: To control the scanner, it is necessary to send a destination coordinate (x_D, y_D) so that it can move to there from its current position (x_C, y_C). If the LIDAR system measures the distance at each position, the number of measurements is equal to the number of positions. Obviously, these numbers depend on how many new positions are sent to the scanner and its maximum scan speed.

First, two-mirror deflections scanners are interfaced by a communication interface (i.e., XY2-100 industrial protocol [12]). An illustration of the protocol is shown in the upper part of Fig. 2. To control two motors, a driver must send 16-bit coordinates via the x and y channels (XCHN, YCHN) for an update. Moreover, the protocol includes four additional control bits to form a 20-bit packet to handle the new position. Hence, it takes 20 cycles to send the destination to the scanners. The specification shows that the clock frequency (CLK) is limited by 2MHz, resulting in a minimal period of 500ns. Eventually, an update requires $10\mu s$ (*i.e.,* $20 \times 500ns$). This implies that the maximum number of updates is limited by 100,000 positions.

Second, the maximum speed of a galvanometer scanner is bounded by the maximum frequency f_{max} at which it is able to finish the travel of $x_{min} \rightarrow x_{max} \rightarrow x_{min}$ in one second. It should be noted that the maximum frequency parameter is usually given in the product specifications. However, the scanning angle is not clearly stated in these specifications. Both $f_{max} = 150$ and $f_{max} = 1000$ are correct if their corresponding scanning angles are not provided. Intuitively, setting a narrower scanning angle makes the scan speed faster. This intuition offers a solution to increase the frame rate by narrowing the scanning FOV.

3) The Maximum Speed of the Scanner: In this subsection, the maximum speed of the galvanometer scanner is formally derived. Due to inertia, updating of the position starts by slowly increasing its speed, reaching to a stable speed, and finally slowly decreases to zero when moving to the destination. This paper assumes that the speed is constant during the movement if it is still under the maximum speed, as this approach does not sacrifice generality.

$$v \leq v_{max} = \frac{2 f_{max}\,(x_{max} - x_{min})}{1s} \tag{6}$$

Here, v_{max} is the maximum average speed of a scanner, and the numerator, $2 f_{max}\,(x_{max} - x_{min})$, is the path length that the scanner traverses in one second.

Example 2: Assume that the galvanometer scanner is able to scan at a maximum frequency $f_{max} = 150$ at an angle of 45° with $x_{max} = 60,000$ and $x_{min} = 0$. Then, its speed is bounded as follows:

$$v \leq v_{max} = \frac{2 \times 150 \times 60,000}{1s} = \frac{18}{\mu s} \tag{7}$$

This indicates that the scanner is not able to move along a path longer than 18 within one microsecond. For example, if each step unit is 12 μrad, the scanner cannot move through an angle longer than 18 step units during that time.

III. Scanning Optimization Problem

Existed double galvanometers are often used in laser marking systems [22]–[24] which are often required to handle an arbitrary pattern. The arbitrary pattern is derived by a complicated algorithm such as a genetic-based approach [24]. This complex scanning pattern is possible because a marking system does not require a fast scanning speed as the marking repetition rate is only about 1kHz to 5kHz [22]. On the other hand, the scanning speed is very important for a LIDAR system so that this paper attempts to find the scanning pattern that increases the scanning speed. To this end, this paper proposes to use a special scanning mode which is a simple yet fast scanning mode. More importantly, our approach gives the optimal scanning pattern for a LIDAR scanning system.

A. Scanning Problem

This subsection addresses the scanning problem of finding the optimal path along which scanners run at the maximum speed. The formal definition of the problem is as follows:

Definition 1 (Scanning Problem): Given a set of N positions in the FOV, the scanning problem is to find the Eulerian trail which visits every position exactly once.

Let $\{p_i\}_{i=1}^{N}$ denote the positions in the FOV. The scanning problem is to find a trail, $q_1 \rightarrow q_2 \rightarrow \ldots \rightarrow q_N$, where the set $\left\{q_j\right\}_{j=1}^{N}$ is equal to $\{p_i\}_{i=1}^{N}$. Although two sets may have identical elements, the orders of their elements can differ. This condition does not indicate that the coordinates of q_i must be identical to those of p_i for any $i = 1, \ldots, N$. In other words, the problem to solve is to find the scanning order in which all positions are visited exactly once. It should be noted that $\left\{q_j\right\}_{j=1}^{N}$ is a permutation of $\{p_i\}_{j=1}^{N}$ and that any such permutation results in a valid order for scanners. Additionally, the condition which holds that every position should be visited exactly once reflects the actual case, in which it is desired to have only one distance measurement at a single position for an image.

When N positions are given, there are $N!$ possibilities to obtain a valid trail for scanners, as $N!$ permutations of $\{p_i\}_{j=1}^{N}$ exist. The natural demand is to find the trail along which the scanners can travel in the shortest time or along the shortest path. Given the assumption that the speed of the scanner is constant during the movement, the traversing time is proportional to the path length. Therefore, the optimal scanning optimization is defined as follows:

Problem 1 (Optimal Scanning Problem): Given a set of N positions $\{p_i\}_{i=1}^{N}$ in the FOV, find the trail $q_1 \rightarrow q_2 \rightarrow \ldots \rightarrow q_N$ that minimizes the total length of the traversing path,

$$L_q = \sum_{\mathrm{i=1}}^{\mathrm{N-1}} ||q_i - q_{i+1}||_2 \,, \tag{8}$$

where the set $\left\{q_j\right\}_{j=1}^{N}$ is equal to $\{p_i\}_{i=1}^{N}$ and where $||.||_2$ denotes a Euclidian distance (i.e., L2-norm), which is formulated as follows:

$$||q_i - q_{i+1}||_2 = \sqrt{(x_i - x_{i+1})^2 + (y_i - y_{i+1})^2} \tag{9}$$

B. An Optimal Scanning Pattern

1) The Grid Graph Representation of the FOV: The FOV is usually represented by a grid graph in which any two consecutive points are aligned at fixed steps. Similar to Section II-A, let $xstep$, $ystep$ denote regular steps on the x-axis and y-axis of the FOV, respectively. Without a loss of generality, it is assumed here that $xstep$ is equal to or smaller than $ystep$. The optimal scanning problem is then to find an optimal trail on the grid graph. First, a feasible solution is considered, i.e., the raster scanning pattern, as shown in the lower part of Fig. 2. Starting from point (x_{min}, y_{min}), scanners visit the next point $(x_{min} + xstep, y_{min})$. They then iteratively visit others along the same line until they reach the right border $(x_{max}- xstep,\ y_{min})$. After going over all points on the first line, the scanners scan the next line in a similar manner. This procedure is repeated until all points on the grid graph are visited. The raster scanning pattern is shown in Alg. 1 and explained in detail in the next subsection.

Algorithm 1: Raster Scanning Algorithm With Laser Scanners

Input: x_{min}, x_{max}, y_{min}, y_{max}, $xstep$, $ystep$
Outputs: x, y
x=x_{min}, y=y_{min}, $dir = 0$
While $\mathrm{y} < \mathrm{y_{max}}$
  /* **Even line** */
  If x==0
    While x<x_{max}- $xstep$
      Measure an object's distance at the scanning angle
      $x := x + xstep$
    End while
    $dir = 1$
  Else
  /* **Odd Line***/
    While x>x_{min}
      Measure an object's distance at the scanning angle
      $x := x$ - $xstep$
    End while
    $dir = 1$
  End if
  $y := y + ystep$
End while

2) An Optimal Scanning Pattern: This subsection proves the optimality of the raster scanning pattern by the proposed theorem, as follows:

Theorem 1 (Optimality): If N positions forming the grid graph in the FOV are given and $xstep$ is equal to or smaller than y$step$, the scanning pattern in Alg. 1 provides an optimal solution to *Problem 1*.

Proof: The scanning pattern derived by Alg. 1 forms a tree because it does not include any cycle. In addition, the tree consists of all vertices in the graph. Therefore, the proof for *Theorem 1* is equivalent to the problem of showing that the tree which is the derived raster scanning pattern is the minimum spanning tree (MST) [25] of the graph. In this paper, the procedure of the well-known Prim algorithm [26] for the MST problem is adopted to address this problem. Let $q_1 = (x_{min}, y_{min})$ be the first node in the MST. In the Prim algorithm, the next node is the candidate of q_2 which is closest to q_1. A candidate solution is the point $(x_{min} + x_{step}, y_{min})$ due to $xstep$ being equal to or smaller than y$step$. The procedure of choosing nodes for the MST on the first line is repeated until the rightmost node $(\mathrm{x_{max}}, \mathrm{y_{min}})$ is reached. Because there is no remaining node in this current line, the next node must be on the second line. According to the Prim algorithm, the MST can be expanded to any node on the second line. However, in order to guarantee the conditions of the MST, the rightmost one $(x_{max}$-$x_{step}, y_{min} + x_{step})$ is selected when its current position

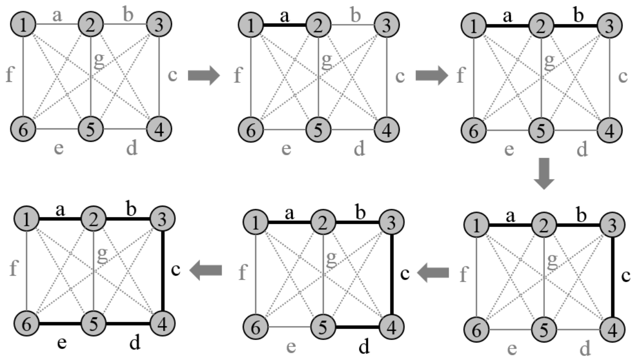


Fig. 3. An example of the Prim algorithm to find a raster scanning pattern on a grid graph.

is (x_{max}- x_{step}, y_{min}). This procedure will be iteratively applied until all nodes in the grid graph are added to the tree to continuously form the MST.

Example 3: Consider an example with six positions, as shown in Fig. 3. Every two nodes are connected by an edge, which is the Euclidean distance between them. The problem is to find the MST over the graph. Here, we push node “1” into the set. Clearly, the minimum edge is “a,” which should be added to the MST, after which node “2” is inserted into the set. The next edge will be the minimum weight edge connected to the set {“1”, “2”}. Therefore, edge “b” is inserted into the MST and node “3” is added to the set. At this point, a minimum edge connected to the set {“1”, “2”, “3”} is required. There are three possible options: “c”, “g” and “f”. Mathematically, it is possible to select any of them, as the graph considered is an undirected graph. However, in reality, it is clear that the current position of the galvanometer is now at node “3.” Eventually, edge “c” is selected and the node “4” is added to the set. Next, edge “d” will be added to the MST similarly and node “5” is inserted into the set. Finally, edge “e” is added to the MST, and the MST is completely constructed.

Although the raster scanning is a series of heuristic steps, it is the optimal solution in this specific case. It should be noted that the assumption that $xstep$ is equal to or smaller than $ystep$ is only used for expressing the proof clearly without a loss of generality. In case $xstep$ is larger than $ystep$, the requirements of Theorem 1 can still be met by slightly modifying Alg. 1 by exchanging the x-axis and the y-axis.

C. Combining the Optimal Sampling Pattern With Timing Constraints

Here, we reconsider the controller in the center of Fig. 2. It requires six parameters: x_{min}, x_{max}, y_{min}, y_{max}, $xstep$ and $ystep$. x_{min}, x_{max}, y_{min} and y_{max} define the FOV of interest, whereas $xstep$ and $ystep$ determine the resolution of the captured image. The goal of this subsection is to investigate the link between the frame rate and the resolution factors.

1) Constraint on Moving Steps: Let t_x and t_y denote the time for scanners to travel the distances of $xstep$ and $ystep$, respectively. The speed of Galvanometer scanners in each small step should satisfy the condition in (6), which states that the maximum speed constraint and thus the speeds of the scanners, $\upsilon_x$and υ_y,, must satisfy the following conditions:

$$\upsilon_x = \frac{xstep}{t_x} \leq \upsilon_{max} = \frac{2f_{max}(x_{max} - x_{min})}{1s} \quad (10)$$

$$\upsilon_y = \frac{ystep}{t_y} \leq \upsilon_{max} = \frac{2f_{max}(y_{max} - y_{min})}{1s} \quad (11)$$

These conditions are defined based on the speed constraints; thus, they can be directly rewritten under the timing constraints, which are derived as follows:

$$\frac{1s}{2f_{max}} \leq \frac{x_{max} - x_{min}}{xstep} \times t_x \quad (12)$$

$$\frac{1s}{2f_{max}} \leq \frac{y_{max} - y_{min}}{ystep} \times t_y \quad (13)$$

Intuitively, the right sides of (12) and (13) indicate the times required for galvanometer scanners to complete the scan of the horizontal and vertical lines, respectively. Therefore, each frame requires the following time,

$$\frac{1}{FPS} = \frac{y_{max} - y_{min}}{ystep} \times \frac{x_{max} - x_{min}}{xstep} \times t_x + \frac{y_{max} - y_{min}}{ystep} \times t_y \quad (14)$$

where FPS denotes the frame rate. The first term on the right side of (14) represents the x-axis scanning time, which is determined by multiplying the number of lines by the horizontal-line scanning time, whereas the second term on the right side of (14) is the vertical-line scanning time. It should be noted that the raster scanning pattern in Alg. 1 is used for this derivation.

2) Two Constraints on the Frame Rate: Based on (12), (13), and (14), the first constraint on the frame rate is derived as follows:

$$\frac{1}{FPS} \geq \left(\frac{y_{max} - y_{min}}{ystep} + 1\right) \times \frac{1s}{2f_{max}} \quad (15)$$

This implies that the frame rate depends on the number of lines in the acquired image. If the scanners scan more lines, $\left(\frac{y_{max}-y_{min}}{ystep}\right)$, FPS becomes smaller. On the other hand, if they scan fewer lines, FPS becomes larger.

Next, the second constraint is derived to compensate for (15), as (15) does not include x-axis parameters which represent the number of pixels on a line (*width*). It should be noted that t_x and t_y are the timing intervals to update the positions on the horizontal and vertical directions, respectively. Therefore, they must be equal to or greater than the timing unit τ by the communication interface (*i.e.,* $t_x \geq \tau$ and $t_y \geq \tau$). Thus, the second condition is directly derived from (14), as follows:

$$\frac{1}{FPS} \geq \left(\frac{y_{max} - y_{min}}{ystep} \times \frac{x_{max} - x_{min}}{xstep} + \frac{y_{max} - y_{min}}{ystep}\right) \times \tau \quad (16)$$

The term on the right of (16) is obtained by multiplying the number of pixels in the image by the time unit for updating the position in the FOV. The inequalities in (15) and (16) provide conditions which link the frame rate, the solution, and the FOV. The following subsection investigates these conditions with numerical examples.

TABLE II
FRAME RATES OVER DIFFERENT RESOLUTIONS OF AN ACQUIRED IMAGE

height	*width*	*FPS* by (15) (fps)	*FPS* by (16) (fps)	*Final FPS* *(fps)*
16	240	17.647	25.934	17.647
16	480	17.647	12.994	12.994
16	640	17.647	9.750	9.750
16	1280	17.647	4.879	4.879
32	240	9.091	12.967	9.091
32	480	9.091	6.497	6.497
32	640	9.091	4.875	4.875
32	1280	9.091	2.440	2.440
64	240	4.615	6.483	4.615
64	480	4.615	3.248	3.248
64	640	4.615	2.438	2.438
64	1280	4.615	1.220	1.220

TABLE III
FOVS OVER DIFFERENT RESOLUTIONS OF AN ACQUIRED IMAGE

height	*width*	FOV_x		FOV_y	
		x_{max}	$angleX_{max}$	y_{max}	$angleY_{max}$
16	240	43,200	29.7°	2,880	**1.98°**
	480	60,000	41.2°		
	640	60,000	41.2°		
	1280	60,000	41.2°		
32	240	43,200	29.7°	5,760	**3.95°**
	480	60,000	41.2°		
	640	60,000	41.2°		
	1280	60,000	41.2°		
64	240	43,200	29.7°	11,520	**7.91°**
	480	60,000	41.2°		
	640	60,000	41.2°		
	1280	60,000	41.2°		

3) Numerical Examples:

Example 4: For the example of the frame rate, the settings in Example 1 in which the galvanometer scanners scan at the maximum frequency ($f_{max} = 150$) and the frame image has the size of 480×240 are reused. Then, (15) can be changed as follows:

$$\frac{1}{FPS} \geq (height + 1) \times \frac{1s}{2f_{max}} \Rightarrow FPS \leq 1.25 \quad (17)$$

Consider the case in which scanners use the XY2-100 industrial protocol [12] for the communication interface. In this case, each update requires $10\mu s$ (*i.e.,* $\tau = 10\mu s$), and (16) is changed as follows:

$$\frac{1}{FPS} \geq (height \times width + height) \times \tau \Rightarrow FPS \leq 0.867 \quad (18)$$

From (17) and (18), the maximal frame rate is determined to be 0.867 because (18) has a smaller inequality criterion. However, (17) only depends on the "*height*" parameter. If the frame width is decreased to 240, (16) is changed as follows:

$$\frac{1}{FPS} \geq (height \times width + height) \times \tau \Rightarrow FPS \leq 1.73 \quad (19)$$

From (17) and (19), the maximal frame rate is determined to be 1.25 because (17) has a smaller inequality criterion. This example clearly shows that either (15) or (16) can be dominant according to the setting of the resolution.

Recall that existing LIDAR systems usually adopt multiple channels of emitters and detectors. Therefore, multiple lines can be scanned simultaneously. The following part investigates the performance of the proposed controller with the same line settings used in the other cases. Consider the Velodyne systems VLP-16 [18], HDL-32E [19], and HDL-64E [20], which simultaneously scan 16, 32 and 64 data lines, respectively. Thus, *height* is set to 16, 32, and 64 for each corresponding system. In Table II, the corresponding frame rates according to different resolutions are presented. It should be noted that the FPS by (15) in the third column is only proportional to *height*; whereas FPS by (16) in the fourth column is proportional to both *width* and *height*. The lower values for these two columns are selected for the final FPS in the fifth column. These results show that the scanners are able to achieve an fps of approximately 17.647 when acquiring an image resolution of 240×16, whereas acquiring an image of 1280×16 leads to a frame rate of 4.879 fps. This outcome indicates that a higher resolution results in a lower frame rate. In other words, the frame rate decreases linearly as the number of scanning lines increases.

Example 5: For the example of the FOV, the FOV factor which is not investigated in Example 3 is considered here. In particular, the frame rate and resolution do not indicate the coverage area in the FOV of interest. One assumption in Example 3 is expressed as follows:

$$t_x = t_y = \tau = 10\mu s \quad (20)$$

This indicates that the update time for the position is 10 μs. Hence, the maximum moving step is 180 due to the constraints on the speed of the scanners. This correlation can be expressed as follows:

$$\max\left(\frac{xstep}{t_x}, \frac{ystep}{t_y}\right) \leq v_{max} = 180 \quad (21)$$

$$\Rightarrow xstep \leq 180 \; and \; ystep \leq 180 \quad (22)$$

Table III shows the FOVs corresponding to the different frame sizes and frame rates. The first and second columns represent the frame *height* and *width*, respectively. From (22), the maximum angles can be derived by multiplying the *width* and *height* with the moving steps. Columns from 3 to 6 show the angles of the FOV. The vertical angle gradually decreases due to the maximum speeds of galvanometer scanners. In practice, the FOV of interest can be extended by increasing t_y in (20). Eventually, y_{step} is increased and the vertical angle is extended accordingly. Note that t_y represents the time unit for updating a scanning line. In general, it only needs *height*-1 times to switch from the current line to the next one. Thus, the FPS only decreases slightly.

IV. A SAMPLE LIDAR SYSTEM

This section presents the sample LIDAR system developed under the proposed concepts.

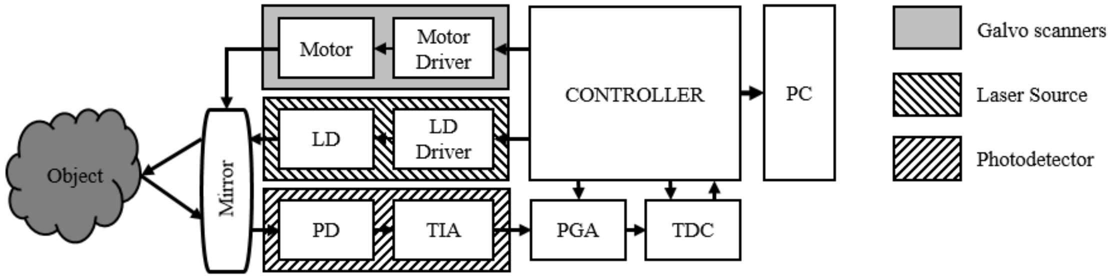


Fig. 4. Block diagram of the prototype LIDAR system.

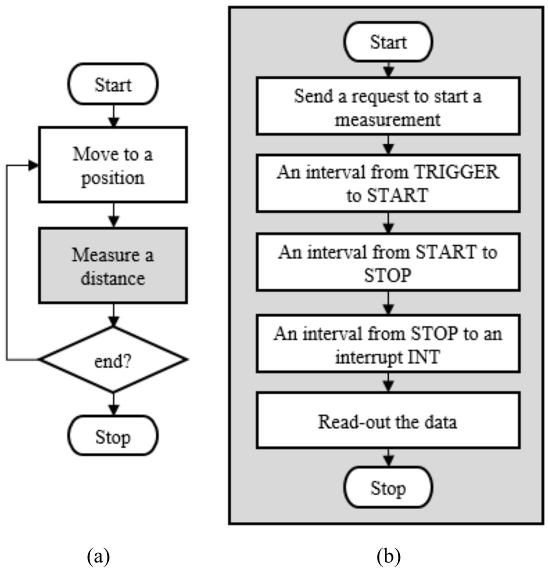


Fig. 5. The procedure to measure distance in the proposed LIDAR system. (a) The main routine. (b) A routine to measure a distance.

A. Verification Framework

In addition to idealistic concepts, it is important to consider an actual verification framework, as it is directly linked to practical applications. Based on this objective, a sample LIDAR system is proposed. Its system block diagram is shown in Fig. 4. The central component of the proposed system is a controller which is built with a combination of a Raspberry PI III board [27] and a LOGI-PI board [28]. The Raspberry PI serves as a high-level communication interface bridge to send the depth measurements from the LOGI-PI board to the display PC via the TCP-IP Ethernet protocol [29]. All low-level interfaces are handled on the FPGA board, LOGI-PI, which uses a low-cost device, a Spartan 6- XC6SLX9-2TQG144C from Xilinx. The controller is connected to the galvanometer scanners (motor and motor drivers) [11], a laser source (i.e., a laser diode, LD), a programmable gain amplifier circuit (PGA), and a time-to-digital converter (TDC) [30]. Signals reflected from objects are captured by a photodetector (PD) [31]. The distance measurement methods [32], [33] are likely to be integrated in our framework. However, it should be noted that this paper concentrates on the interface between the controller and the galvanometer scanners. The selection of elements of other optical elements and the timing measurement circuits may be different, which is out of the scope of this paper.

TABLE IV
PARAMETER SETTINGS WITH THE PROPOSED SAMPLE LIDAR SYSTEM FOR A GIVEN FRAME RATE

Frame rate	Width	Height	Number of pixels per frame	Number of measurements
20	360	12	4,320	86,400
10	360	27	9,720	97,200
5	360	54	19,440	97,200
2	360	135	48,600	97,200
1	360	270	97,200	97,200
0.5	450	440	198,000	99,000

Fig. 5 illustrates the main routine of measuring the distances. The raster scanning algorithm in Alg. 1 is simplified in Fig. 5(a), in which the measurement of a single point is conducted by iteratively mapping the position to the measured distance. Each measurement starts by triggering an LD to emit light at the position and also enabling a TDC by the START signal. It should be noted that the emitted light comes to the object and that a reflected signal returns to the system and reaches the detector PD (referred to as a STOP signal). The TDC then measures the time interval between two signals, which is read out to the controller and translated into the distance. This procedure is summarized in Fig. 5(b).

In this system, considering the interface between the controller and the galvanometer scanner, the trade-off between the resolution and the frame rate should be addressed. These are explained in the following subsections.

B. Speed Evaluation

This subsection discusses experiments with the proposed LIDAR system at different frame rates. The results of the experiments are presented in Table IV. The proposed LIDAR system achieves various frame rates by setting the corresponding frame size. For example, from the second row to the sixth row, the *width* in the second column is fixed at 360, but the *height* in the third column is adjusted to vary the frame rates in the first column to 20, 10, 5, 2, and 1. The fourth and last columns present the number of pixels per frame and the number of measurements, respectively. It should be noted that the number of measurements must be equal to or less than 100,000, which is the maximum number of updates.

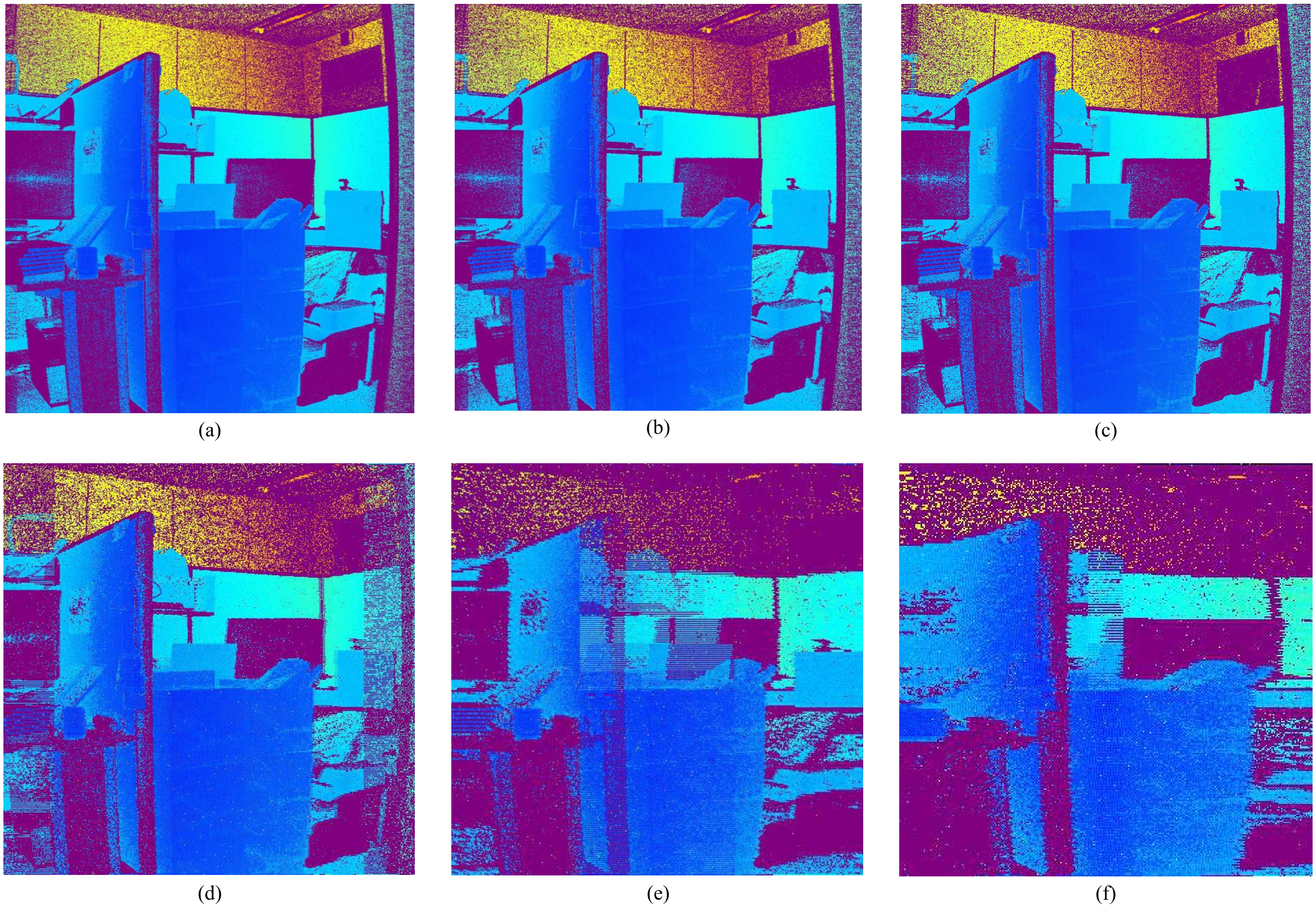


Fig. 6. Images according to various resolutions by the proposed sample LIDAR system.

The results show that the proposed system can achieve nearly 100,000 measurements per second despite the fact that it only uses one emitter and detector pair. On the other hand, the Velodyne LIDAR sensors VLP-16, HDL-32E, and HDL-64E, which use 16, 32, and 64 channels, respectively, correspondingly achieve 300,000, 700,000, and 2,200,000 measurements per second. These results indicate that each channel for these sensors only provides approximately 20,000 measurements per second. Therefore, the proposed system shows much better performance and is also able to adjust the parameters to fit the desired frame rate. In addition, the scanning speed is likely to increase when the weight and size of galvanometer scanners are reduced. In particular, the galvanometer scanner can be further reduced in size as that in [34], whose volume is smaller than $4 \times 3 \times 2$ (cm^3).

C. Subjective Evaluation

Fig. 6 shows the experimental images provided by the proposed sample LIDAR system. Six images were taken at various resolutions of 600×600, 500×500, 400×400, 300×300, 200×200, and 150×150. The results show that the images 600×600, 500×500, and 400×400 in size nicely capture the same FOV and clearly identify the objects at different distances. Meanwhile, as the resolutions decrease, the FOVs are gradually narrowed in the captured images of 300×300, 200×200 and 150×150. These images also contain boundary artifacts which are caused by the mapping error. These phenomena can be explained as follows. In a LIDAR system, it is important to map a position to a distance, which is depicted in Fig. 5(a). The mapping error occurs when the time synchronization between the scanners and measurement module is violated. Consider the images of 400×400 and 300×300. If the FOV is $60,000 \times 60,000$, the moving steps will be 150 and 200, respectively. Recall the speed constraint in (22), which is derived from (6) and (7). Therefore, the moving step when obtaining a 400×400 image encounters (2), whereas the step when obtaining a 300×300 image does not. In practice, the scanners cannot exceed the maximum speed. Eventually, the FOV decreases. Furthermore, boundary artifacts arise because the scanners change the scanning directions in those areas and then require more time for scanning. In summary, the images show the visual impact of the derivations in this paper. The message is that violating the conditions leads to a narrowing of the FOV and to boundary artifacts. In other words, derivations in (22) provide a sufficient condition to achieve a “good” image.

D. Ranging Accuracy Evaluation

This subsection briefly discusses the accuracy of our LIDAR system based on two criteria: the vertical resolution

TABLE V
VERTICAL RESOLUTION COMPARISON OF VARIOUS LIDAR SYSTEMS

Model	Numbers of channels	Vertical Angle	Vertical Resolution
IBEO ALASCA [16]	4	3.2º	0.8º
LEDDAR 16M [17]	16	×	×
PUCK^{TM} VLP-16 [18]	16	+15º to -15º	2º
HDL-32E [19]	32	+10º to -30º	1.33º
HDL-64E [20]	64	26.9º	0.4º
Riegl-VUX-1UAV [21]	1	×	×
Our LIDAR system	**1**	**41.2º**	**up to 0.0007º**

TABLE VI
MEAN/DEVIATION OF DISTANCE MEASUREMENTS AT DIFFERENT POSITIONS AND RESOLUTIONS

Image	Pos. 1 (m)	Pos. 2 (m)	Pos. 3 (m)	Pos. 4 (m)	Pos. 5 (m)	Pos. 6 (m)
600×600	3.409/ 0.037	3.481/ 0.048	2.934/ 0.046	2.982/ 0.052	2.246/ 0.072	2.088/ 0.043
500×500	3.432/ 0.035	3.493/ 0.048	2.927/ 0.056	3.011/ 0.060	2.226/ 0.065	2.078/ 0.062
400×400	3.436/ 0.041	3.469/ 0.068	2.959/ 0.057	3.019/ 0.049	2.266/ 0.068	2.101/ 0.089

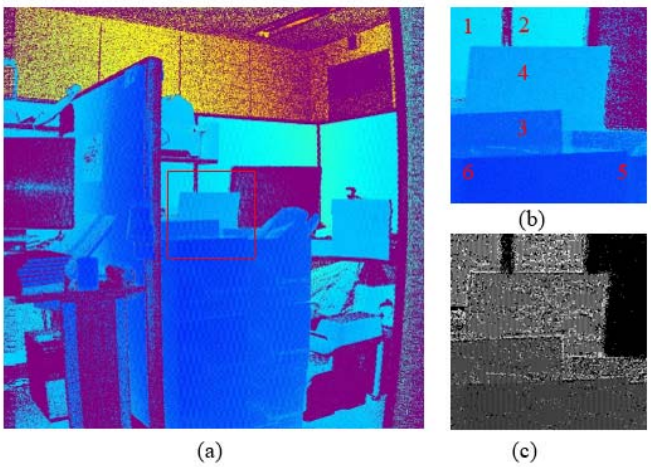


Fig. 7. An experiment setup for the ranging accuracy estimation. (a) The center-cropped position. (b) The cropped region of interest and positions. (c) The corresponding map.

and ranging accuracy. The vertical resolution is an important factor to define an object for general object detection applications. Table V reports the vertical resolutions of various LIDAR systems including our LIDAR prototype. The results show that our system outperforms the existing systems in terms of vertical resolution. By utilizing the high resolution of galvanometers, our LIDAR system can achieve 12μrad (=0.0007° vertical resolution. Moreover, the vertical resolution is likely to be adjusted for specific applications as shown in Section III-C. In particular, the resolutions are configured to achieve the desired frame rate and FOV.

The evaluation for the ranging accuracy of our LiDAR system is conducted on the obtained images in Fig. 6 as follows. First, a box at center or close points is cropped. Second, six positions in the small box are selected and sixes 8×8 corresponding blocks are taken as shown in Fig. 7. Finally, for each small block, the mean and deviation are calculated. This simple experiment makes an assumption that distances in a small block are similar. Therefore, each position is marked with a mean and deviation which are considered to the measured distance and error, respectively. The detailed results are reported in Table VI. For each position, its mean and deviation of distances are reported. For example, the position 1 in the image 600×600 is at 3.407m with the error is 0.037m (or 3.7cm). On average, the error is about 0.055m or 5.5cm. To this end, the experimental results show that the average error is about 5.5cm, which is likely to be suitable for many practical applications, e.g., autonomous driving cars.

V. CONCLUSION

This paper presents a simple model of a scanner controller in a LIDAR system and provides an optimal solution for a scanning problem. The relationships among the frame rate, the resolution and the FOV are investigated using the proposed model. While conditions in (6) and (21) give sufficient constraints to achieve a good image quality, the inequality criteria in (15) and (16) provide a means of estimating the frame rate. Furthermore, a sample LIDAR system which adopts the proposed analyses achieves scalable frame rates and resolutions.

REFERENCES

[1] S. Thrun *et al.*, "Stanley: The robot that won the DARPA grand challenge: Research articles," *J. Robot. Syst.*, vol. 23, no. 9, pp. 661–692, 2006.
[2] C. Urmson *et al.*, "Autonomous driving in urban environments: Boss and the Urban Challenge," *J. Field Robots*, vol. 25, no. 8, pp. 425–466, 2008.
[3] J. Levinson *et al.*, "Towards fully autonomous driving: Systems and algorithms," in *Proc. 4th IEEE Intell. Veh. Symp.*, Jun. 2011, pp. 163–168.
[4] E. Ackerman, "Lidar that will make self-driving cars affordable [news]," *IEEE Spectr.*, vol. 53, no. 10, p. 14, Oct. 2016.
[5] M. D. Adams, "Coaxial range measurement—Current trends for mobile robotic applications," *IEEE Sensors J.*, vol. 2, no. 1, pp. 2–13, Feb. 2002.
[6] H. Deilamsalehy and T. C. Havens, "Sensor fused three-dimensional localization using IMU, camera and LiDAR," in *Proc. IEEE Sens.*, Nov. 2016, pp. 1–3.
[7] G. Zhou, Z. Zhou, J. Yang, Y. Tao, X. Nong, and O. Baysal, "Flash lidar sensor using fiber-coupled APDs," *IEEE Sensors J.*, vol. 15, no. 9, pp. 4758–4768, Sep. 2015.
[8] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? The KITTI vision benchmark suite," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2012, pp. 3354–3361.
[9] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The KITTI dataset," *Int. J. Robot. Res.*, vol. 32, no. 11, pp. 1231–1237, 2013.
[10] D. S. Hall, "High definition LiDAR system," U.S. Patents 7 969 558 B2, Jun. 28, 2011.
[11] Sino-Galvanometer Technology Co., Ltd, Beijng, China. (2015). *SG8220 Galvanometer Scanner*. [Online]. Available: http://www.sino-galvo.com/ypros-show-124-488.aspx
[12] Newson Engineering, Dendermonde, Belgium. (2007). *Rhothortm X7 I/O Configuration: XY2-100 Technical Datasheet*. [Online]. Available: http://www.newson.be/doc.php?id=XY2-100
[13] K. Konolige, J. Augenbraun, N. Donaldson, C. Fiebig, and P. Shah, "A low-cost laser distance sensor," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2008, pp. 3002–3008.

[14] Y. Li, “Beam deflection and scanning by two-mirror and two-axis systems of different architectures: A unified approach,” *Appl. Opt. J.*, vol. 47, no. 32, pp. 5976–5985, 2008.

[15] SCANLAB AG, Puchheim, Germany. (2006). *The RTC 4 PC Inteface Board for Real Time Control of Scan Heads and Lasers*. [Online] Available: rlab.org.uk/mediawiki/images/3/3e/RTC4.pdf

[16] J. Scholz, V. Willhoeft, D. R. Schulz, and T. Kluge. (2006). ALASCA user manual, Ibeo Automotive Systems, Hamburg, Germany. [Online]. Available: http://www.raginbot.com/wiki/images/d/d8/Manual_ALASCA.pdf

[17] LeddarTech Inc, Québec, Canada. (May 2017). *LEDDAR M16: Multi-Element Sensor Module*. [Online]. Available: http://leddartech.com/modules/m16-multi-element-sensor-module/

[18] Velodyne LiDAR Inc, Morgan Hill, CA, USA. (2017). *Velodyne LiDAR PuckTM: Real-Time 3D LiDAR Sensor*. [Online]. Available: http://velodynelidar.com/vlp-16.html

[19] Velodyne LiDAR Inc, Morgan Hill, CA, USA. (2017), *Velodyne LiDAR HDL-32E: High Definition Real-time 3D LiDAR Sensor*. [Online]. Available: http://velodynelidar.com/hdl-32e.html

[20] Velodyne LiDAR Inc, Morgan Hill, CA, USA. (2017), “*Velodyne LiDAR HDL-64E S3: High Definition Real-time 3D LiDAR Sensor*. [Online]. Available: http://velodynelidar.com/hdl-64e.html

[21] RIEGL USA Inc, Orlando, FL, USA. (2017). *VUX-1UAV: Unmaned Laser Scanners for UAS/UAV-Based Laser Scanning*. [Online]. Available: http://products.rieglusa.com/product/all-categories-unmanned-scanners/vux-1uav-survey-grade-unmanned-laser-scanners

[22] M.-F. Chen *et al.*, “A scribing laser marking system using DSP controller,” *Opt. Lasers Eng.*, vol. 46, no. 5, pp. 410–418, 2008.

[23] W. Dongyun and Y. Xinpiao, “An embedded laser marking controller based on ARM and FPGA processors,” *Sci. World J.*, vol. 2014, 2014, Art. no. 716046. [Online]. Available: https://www.hindawi.com/journals/tswj/2014/716046/abs/, doi: 10.1155/2014/716046.

[24] D. Wang, Q. Yu, and Y. Zhang, “Research on laser marking speed optimization by using genetic algorithm,” *PLoS ONE*, vol. 10, no. 5, p. e0126141, 2015.

[25] R. L. Graham and P. Hell, “On the history of the minimum spanning tree problem,” *Ann. History Comput.*, vol. 7, no. 1, pp. 43–57, Jan. 1985.

[26] R. C. Prim, “Shortest connection networks and some generalizations,” *Bell Syst. Tech. J.*, vol. 36, no. 6, pp. 1389–1401, 1957.

[27] *Raspberry Pi board*. Accessed: Jul. 2, 2017. [Online] Available: https://www.raspberrypi.org/

[28] *LOGI-PI FPGA Board*. Accessed: Aug. 27, 2017. [Online] Available: http://valentfx.com/logi-pi/

[29] L. Parziale *et al.*, “Application structure and programming interfaces,” in *TCP/IP Tutorial and Technical*, 8th ed. Armonk, NY, USA: IBM Redbooks, 2006. [Online]. Available: http://www.redbooks.ibm.com/abstracts/gg243376.html?Open

[30] Texas Instruments Inc. (2017). *TDC7200: Time-to-Digital for Time-of-Flight Applications in LiDAR, Magnetostrictive and Flow Meters*. [Online]. Available: http://www.ti.com/product/TDC7200

[31] Thorlabs Inc., Newton, NJ, USA. (2017). *Avalanche Photodetector: APD110x/120x Operation Manual*. [Online]. Available: https://www.thorlabs.com/thorproduct.cfm?partnumber=APD110C

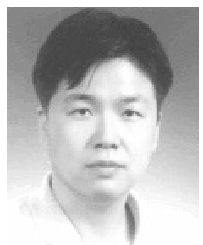
[32] T. Fersch, R. Weigel, and A. Koelpin, “A CDMA modulation technique for automotive time-of-flight LiDAR systems,” *IEEE Sensors J.*, vol. 17, no. 11, pp. 3507–3516, Nov. 2017.

[33] D. Perenzoni, L. Gasparini, N. Massari, and D. Stoppa, “Depth-range extension with folding technique for SPAD-based TOF LIDAR systems,” in *Proc. IEEE Sens.*, Nov. 2014, pp. 622–624.

[34] New Scale Technologies Inc, Victor, NY, USA. (Aug. 30, 2017). *Two-Axis/Two-Mirror Beam Steering System*. [Online]. Available: https://www.newscaletech.com/wp-content/uploads/DK-M3-RS-U-2M-20-L-datasheet.pdf

Xuan Truong Nguyen received the B.S. degree in electrical engineering from the Hanoi University of Science and Technology, Hanoi, Vietnam, in 2011, and the M.S. degree in electrical engineering and computer science from Seoul National University, Seoul, South Korea, in 2015. He is currently pursuing the Ph.D. degree in electrical engineering and science with Seoul National University. His research interests are in the areas of algorithm and VLSI design for multimedia applications.

Van Luan Dinh received the B.S. degree in computer science from the Hanoi University of Science and Technology, Hanoi, Vietnam, in 2013. He is currently pursuing the M.S. degree in electrical engineering and computer science with Seoul National University, Seoul, South Korea. His research interests are in the areas of VLSI design for multimedia applications.



Hyuk-Jae Lee received the B.S. and M.S. degrees in electronics engineering from Seoul National University, South Korea, in 1987 and 1989, respectively, and the Ph.D. degree in Electrical and Computer Engineering from Purdue University, West Lafayette, IN, in 1996.

From 1998 to 2001, he was with the Server and Workstation Chipset Division, Intel Corporation, Hillsboro, OR, as a Senior Component Design Engineer. From 1996 to 1998, he was with the Faculty of the Department of Computer Science, Louisiana Tech University, Ruston, LS. In 2001, he joined the School of Electrical Engineering and Computer Science, Seoul National University, South Korea, where he is currently a Professor. He is a Founder of Mamurian Design, Inc., a fabless SoC design house for multimedia applications. His research interests are in the areas of computer architecture and SoC design for multimedia applications.

Hyun Kim received the B.S., M.S., and Ph.D. degrees in electrical engineering and computer science from Seoul National University, Seoul, South Korea, in 2009, 2011, and 2015, respectively.

Since 2015, he was with BK21 Creative Research Engineer Development for IT, Seoul National University, Seoul, South Korea, where he is currently an Assistant Professor. His research interests are in the areas of low-power algorithm, computer architecture, and SoC design for multimedia applications.