

# Fusing Laser and Vision Data with a Genetic ICP Algorithm

Quirin Mühlbauer, Kolja Kühnlenz and Martin Buss  
Institute of Automatic Control Engineering (LSR)  
Technische Universität München  
D-80290 München, Germany  
{qm, koku, mb}@tum.de

**Abstract**—Knowledge about the environment is essential for humanoid and mobile robots to move and act safely. The most intuitive way to perceive information about the environment is through the vision system. However, the accuracy provided by stereo vision is insufficient for many tasks. A more accurate representation is created by a laser range-finder, which delivers no color information. This paper describes a novel approach to merge data obtained by stereo vision and laser range-finders by using a genetic ICP algorithm, which is able to register noisy point clouds with different resolutions and a small overlap. Furthermore, it describes an easy-to-use and robust method to calibrate the extrinsic parameters of two or more laser range-finders.

## I. INTRODUCTION

A robot has many possibilities to observe the environment. Next to the vision system, another biological inspired technology is the use of ultrasound, which delivers a cheap and easy-to-use sensor. A more accurate map is created by a laser range-finder, which is more expensive and needs more energy. Every type of scanner has advantages and disadvantages, a laser range-finder provides the best spatial resolution, but no color information. However, a high accuracy is essential for a robot to act safely and color information is very helpful to recognize and classify objects and thus for the creation of a detailed semantic network. Consequently, these three types and any combinations of them are widespread. The fusion of laser and vision data will yield in a representation of the environment that delivers both, an accurate and colored map which can be used for navigation and cognitive understanding of the environment.

Sonar and laser range-finders are built to deliver depth-information with no color information, whereas a single camera delivers a two dimensional colored picture. To obtain depth information by a vision system, two or more calibrated (the intrinsic like focal length, principal point, skew coefficient and the distortion coefficients and the extrinsic camera parameters like the position and orientation in space have to be known) cameras must be used. A adequate stereo matching algorithm creates a depth map using pictures of the camera. This depth information can be used to create a rough map of the environment, represented by a colored point cloud. In order to enhance the accuracy of the colored point cloud, it can be fused with the data delivered by a laser-range finder. This will yield into a highly precise colored point cloud.

The remainder of this paper is organized as follows: An outline of the state of the art will be given in section 2. The algorithm used for the calibration of the laser range-finders will be described in section 3, while section 4 includes the genetic ICP algorithm and section 5 some experimental results. A last, a conclusion and outlook to future works will be given in section 6.

## II. STATE OF THE ART

Before laser and vision data can be fused, the extrinsic parameters (i.e. the position and orientation) of the laser range-finders have to be measured. As the alignment of the laser scanners is changed frequently, the measurement should be automated. This procedure is called *calibration*. Previous work deals mostly with the calibration of the intrinsic parameters of a laser range-finder [2] and the calibration between a laser range-finder and a camera [10], [18] and are specialized for a single setup of the system and type of laser range-finder. A common approach, which is easy to use and can calibrate all types of laser range-finders automatically still has to be developed.

When it comes to registration of two or more scans, the *iterative closest point* algorithm [1] is widely spread. There exist many different types of the algorithm. Most common use quaternions [1], [4] or singular value decomposition [17] to compute the transformation between the point clouds. Other algorithms use standard optimization methods [7], [11] to minimize an error function. Another approach is the use of extended gaussian images [3], [9], an alternative representation of the shape of surfaces which can easily be computed. An overview about other ICP-variants which use different methods to optimize the execution time [5] can be found in [14]. The main disadvantage of the ICP-algorithm is its disability to align point clouds with different resolutions, noise or with a small overlap. So far, no algorithm which can register those point clouds properly and reliable has been developed.

If the measured data cannot be fused in a proper way, the usage of multiple types of sensors is in vain. Previous work has dealt with the fusion of image data and point clouds created with a laser range-finder. The most common application is the creation of an accurate textured reconstruction of indoor or outdoor scenes [8], [16]. Other applications are localization [12], object scanning [15] and gaining semantic knowledge

[13]. These algorithms transform the point cloud measured by the laser to the cameras point of view and perform a ray based mapping of color information. If the transformation is not correct, this will yield to displacements and distortions and the whole scan is incorrect. Although it is an obvious and promising approach, the use of stereo image processing is not common. Possible errors will only have local affects, and the rest of the scan can be used for further processing.

### III. CALIBRATION OF TWO LASER RANGE-FINDERS

The area seen by one laser range-finder may be too small, so it is useful to use more than one, each providing one point cloud. Furthermore, using two laser range-finders will yield to a higher accuracy. To create a single accurate point cloud, these point clouds have to be merged. For this, the exact position and orientation of each laser has to be know. As the setup of the lasers is changed frequently, the calibration progress should be automated. We created a fast, easy-to-use and reliable algorithm to perform the calibration of most of the extrinsic parameters, the intrinsic parameters are dependent on the configuration of the range-finder and assumed to be known. The algorithm uses the scan of an object with a known height and calculates the following parameters:

- The absolute  $y$  and  $z$  position of the scanner.
- For one laser, the  $x$  position has to be known. The  $x$  position of the other range-finders will be computed relatively to this position.
- The absolute yaw- ( $\Psi$ ), pitch- ( $\Theta$ ) and roll ( $\Phi$ ) angle of the orientation.

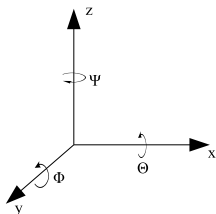


Fig. 1. The extrinsic parameters

The extrinsic parameters are shown in figure 1. Each sensor has one attached coordinate system. The  $x$ -axis is defined by the horizontal axis of the image plane and the  $z$ -axis by the vertical axis. Hence, the  $y$  axis points out of the image plane. The reference system attached to the robot is located in its center of rotation, so the robot will rotate around the  $z$ -axis and drive in direction of the  $y$ -axis. Figure 2 shows the setup of the mobile robot  $R$  with two laser range-finders  $L_1$  and  $L_2$ . Both scanners have a  $\Phi$  of  $45^\circ$  respectively  $-45^\circ$  compared to the reference system. The areas  $A_1$  and  $A_2$  denote the scanning area. When the mobile Robot is rotated around the  $z$ -axis, the whole calibration object  $C$  can be scanned. Furthermore, the stereo camera  $S$  is depicted.

#### A. Requirements

Before the calibration is performed, an initial guess for  $\Phi$  of each laser range-finder should be made. This will increase the

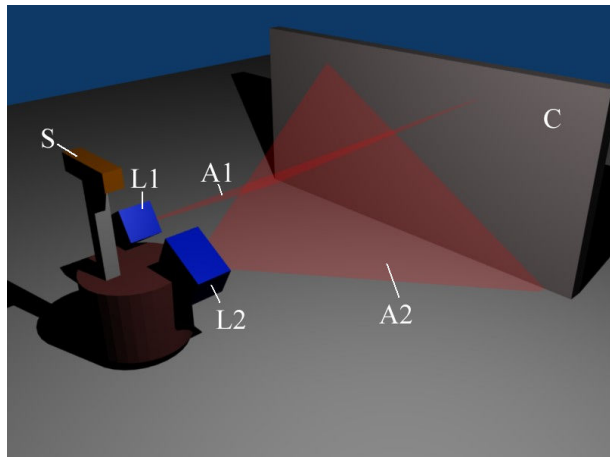


Fig. 2. Setup of a calibration-scene

execution time and avoid absurd results. For the initial guess an accuracy of about  $\pm 45^\circ$  is adequate. The  $x$ -position of the first laser should be set. If it is left to zero, inaccurate results may occur. An object must be scanned using all laser range-finders. The height of the object  $h_O$  must be known, the base point of the object must lay on the floor (where  $z = 0$ ) and the floor has to be flat, so that no scans below the  $x/y$ - plane can occur. After the scan has been performed, the object should be extracted from the single laser scans. Figure 3 shows the laser

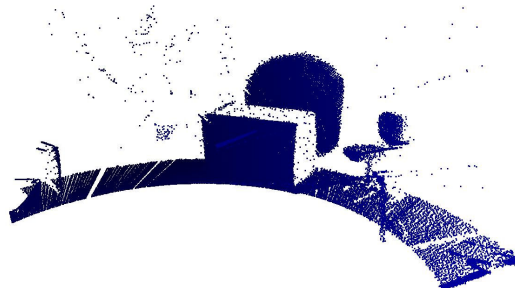


Fig. 3. Scan of the calibration-scene

scan of the calibration scene described above. In the center of the picture the table can be seen, which has to be extracted in a nex step. The scanned object behind the calibration object is a part of the wall. The object on the right side is a chair.

#### B. Algorithm

After the scans have been made, the algorithm is executed in two steps:

1) *The calibration for each laser range-finder:* As described in algorithm 1, the  $z$ -position and  $\Phi$  of the single laser range-finders are computed iteratively in the first step. Changing these parameters will result in a warped representation of the scanned object. The algorithm is complete, when the minimal error  $\epsilon$  is reached. In a real environment  $\epsilon$  should be in the same order of magnitude than the resolution of the laser range-finders, so it is sufficient to set  $\epsilon = 0.1mm$ .

---

**Algorithm 1** Calibration

---

```
1: for each scan do
2:   while  $|z_{min}| > \epsilon$  and  $||z_{max} - z_{min}| - h_O| > \epsilon$  do
3:     Calculate  $z_{min}$  and  $z_{max}$  of the scanned object
4:     if  $z_{min} < 0 + \epsilon$  then
5:       increase  $z$ 
6:     end if
7:     if  $z_{min} > 0 - \epsilon$  then
8:       decrease  $z$ 
9:     end if
10:    if  $(z_{max} - z_{min}) < 0 + \epsilon$  then
11:      increase  $\Phi$ 
12:    end if
13:    if  $(z_{max} - z_{min}) > 0 - \epsilon$  then
14:      decrease  $\Phi$ 
15:    end if
16:  end while
17: end for
```

---

2) *The calibration of all laser range-finders:* After the single laser range-finders have been calibrated, the positions of the scanners relative to themselves can be computed. To calculate the alignment, the first scanner with known  $x$ -position is handled as reference. The position and orientation of the other scanners will be computed relative to the reference scanner. For the calculation of the  $x$ - and  $y$ -position, as well as  $\Psi$  and  $\Theta$ , our ICP algorithm (a detailed description will be given in section IV) is used. If the  $y$ -position of the reference laser is not known, the  $y$ -axis will be placed between the laser range-finders.

#### IV. REGISTRATION OF VISION AND LASER DATA

The ICP algorithm is widely spread to match two point clouds. It estimates the translation and rotation relatively between two point clouds  $C_1$  and  $C_2$  and is repeated iteratively.  $C_1$  has  $n_1$  points,  $C_2$   $n_2$  respectively. It can be described in the following way:

- 1) Allocate for each point  $\mathbf{p}_i^1$  of  $C_1$  the closest point  $\mathbf{p}_i^2$  in  $C_2$ .
- 2) Compute  $\mathbf{R}$  and  $\mathbf{t}$  to minimize the median error  $e(\mathbf{R}, \mathbf{t})$  in equation 1.

$$e(\mathbf{R}, \mathbf{t}) = \frac{1}{n_1} \sum_{i=0}^{n_1} \|\mathbf{p}_i^1 - (\mathbf{R} \cdot \mathbf{p}_i^2 + \mathbf{t})\|^2 \quad (1)$$

- 3) Transform  $C_2$  with the calculated  $\mathbf{R}$  and  $\mathbf{t}$ .
- 4) Iterate until  $e(\mathbf{R}, \mathbf{t})$  converges.

ICP works fine for similar point clouds with a large overlap and a good guess of the initial transformation. In an ideal case, it converges in less than 10 iterations. If the initial guess is not good enough, there is a risk that the algorithm only finds a local minimum. Another problem will occur, if the overlap of the two scans is not large enough. The algorithm has to determine, which points of the two clouds overlap and use only those overlapping points for the registration. The point

cloud, containing only the overlapping points  $\tilde{\mathbf{p}}_i^x, i \in 0 \dots n_x$  is denoted as  $\tilde{C}_x$ .

#### A. Computation of the Transformation

Since it has some advantages compared to a SVD (singular value decomposition) based algorithm in two and three dimensional spaces, our algorithm uses a quaternion based ICP algorithm as described in [1] and [4]. This algorithm uses a cross-covariance matrix to solve the least square problem in equation 2.

$$\underset{\mathbf{R}, \mathbf{t}}{\operatorname{argmin}} = \frac{1}{\tilde{n}_1} \sum_{i=0}^{\tilde{n}_1} \|\tilde{\mathbf{p}}_i^1 - (\mathbf{R} \cdot \tilde{\mathbf{p}}_i^2 + \mathbf{t})\|^2 \quad (2)$$

In this equation only those points are considered, where an overlap is expected.  $\tilde{C}_1$  includes all points, which fulfill equation 3.

$$0 \leq d(\mathbf{p}_i^1, \mathbf{p}_i^2) \leq \frac{d_{max} - d_{min}}{2}, \quad (3)$$

$$d(\mathbf{p}_i^1, \mathbf{p}_i^2) = \|\mathbf{p}_i^1 - \mathbf{p}_i^2\|^2$$

Where  $d_{max}$  denotes the maximal distance between two points and  $d_{min}$  the minimal distance.

#### B. The Genetic ICP Algorithm

As a standard ICP algorithm wasn't able to align vision and laser scans properly, we used a hybrid algorithm to find the global minimum of equation 2. Our algorithm consists out of two main parts, a standard ICP algorithm and a genetic algorithm. Every time the standard ICP algorithm converges against a value, the genetic algorithm is activated to find alternative solutions. The genetic algorithm performs both, random mutations and reproduction of two individuals. Each individual is thereby described by six parameters:  $x$ -,  $y$ - and  $z$ -position as well as  $\Psi$ ,  $\Theta$  and  $\Phi$ . Before the genetic ICP algorithm is executed, the point clouds are clustered with the algorithm described in [6]. To remove small misplaced points, which do not belong to any object, clusters which have less than a minimum number of points are discarded.

The algorithm is described in algorithm 2, where  $i$  denotes the generation and  $s$  the size of the population. In the first step, the overlap is computed for both point clouds and the standard ICP algorithm is applied. The genetic mutation will not be performed every generation, but only if the distance measurement  $d_{i-1}$  converges, i.e. the difference between  $d_{i-1}$  and  $d_{i-2}$  falls below a threshold. Algorithm 2, lines 6 to 15 describe the genetic mutation ( $\operatorname{rand}(x)$  returns a random integer between 1 and  $x$ ,  $\operatorname{rand}2(x)$  a floating number between  $-x$  and  $x$ ), where parameter  $p = 1$  denotes the  $x$ -position,  $p = 2$  the  $y$ -position and so on. By changing the allocation between  $p$  and a parameter, the algorithm can easily be limited to search in two arbitrary dimensions. Furthermore, it is possible to lock arbitrary parameters.

After the transformation of  $C_1$ , the distance measurement  $d_i$  for the iteration  $i$  will be computed. If  $d_i$  is smaller than the previous best distance measurement  $d_{best}$ , the transformation

---

**Algorithm 2** Genetic algorithm
 

---

```

1: reset  $d_{best}$ 
2:  $i = 0$ 
3: while  $i < i_{max}$  do
4:    $i++$ 
5:   Compute  $\tilde{C}_1, \tilde{C}_2$  and corresponding points
6:   Compute  $\mathbf{R}_i, \mathbf{t}_i$  with ICP
7:   if  $|d_{i-2} - d_{i-1}| < \epsilon$  then
8:      $v = rand(s)$ 
9:     load individual  $I_v$ 
10:     $w = rand(s)$ 
11:    load individual  $I_w$ 
12:     $p = rand(6)$ 
13:    substitute parameter  $p$  of  $I_v$  with parameter  $p$  of  $I_w$ 
14:     $p' = rand(6)$ 
15:     $f = rand2(f_{max})$ 
16:    increase parameter  $p'$  of  $I_v$  by  $f$ 
17:    load  $\mathbf{R}_i$  and  $\mathbf{t}_i$  from individual  $I_v$ 
18:  end if
19:  transform  $C_1$  with  $\mathbf{R}_i$  and  $\mathbf{t}_i$ 
20:  Compute distance measurement  $d_i$ 
21:  if  $d_i < d_{best}$  then
22:    save  $\mathbf{R}_{best}$  and  $\mathbf{t}_{best}$ 
23:     $d_{best} = d_i$ 
24:    if  $s < s_{max}$  then
25:      add  $\mathbf{R}_i$  and  $\mathbf{t}_i$  as individual  $I_i$  to population
26:       $s++$ 
27:    else
28:      remove oldest individual from population
29:      add  $\mathbf{R}_i$  and  $\mathbf{t}_i$  to population
30:    end if
31:  else
32:    discard  $\mathbf{R}_i$  and  $\mathbf{t}_i$ 
33:  end if
34:  Return  $\mathbf{R}_{best}$  and  $\mathbf{t}_{best}$ 
35: end while

```

---

$\mathbf{R}_i$  and  $\mathbf{t}_i$  will be stored and added as a new individual to the population. If the maximal number of individuals in the population is reached, the oldest individual will be replaced. If  $d_i$  is larger than  $d_{best}$ , the the transformation  $\mathbf{R}_i$  and  $\mathbf{t}_i$  will be discarded.

To minimize the mean square error while maximizing the points used for the matching, the distance measurement has to be chosen deliberately. Equation 4 tries to fulfill these conditions.

$$d_i = \left( 2 - \left( \frac{\tilde{n}_1}{n_1} \right)^2 \right) \cdot \left( 1 + \left( \frac{1}{\tilde{n}_1} \sum_i d(\mathbf{P}_i^1, \mathbf{P}_i^2) \right)^{\frac{1}{2}} \right) \quad (4)$$

$\tilde{n}_1$  denotes the number of points used for matching. The relative number of points used is denoted as  $\frac{\tilde{n}_1}{n_1}$ . The radicand denotes the median distance between corresponding points. Figure 4 shows a plot of the distance measurement. The absolute minimum is with  $\frac{\tilde{n}_1}{n_1} = 1$  and a median distance of

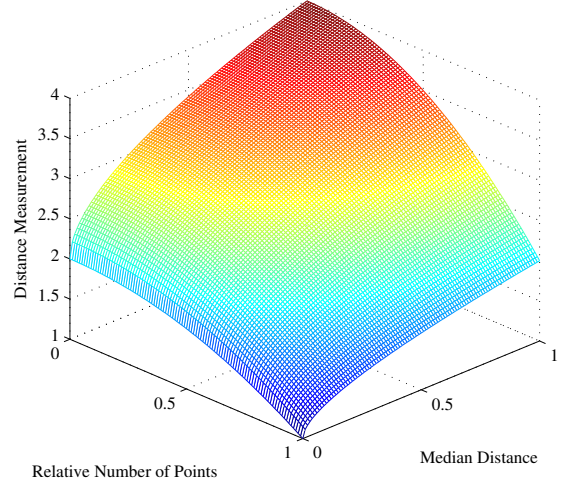


Fig. 4. Plot of the distance measurement depending on the median distance and the relative number of points used

0. When one of the values is increased, the resulting distance will increase as well. Increasing both of them, will lead to the absolute maximum of our distance measurement. By changing the exponents, the weight can be shifted between number of points used and the median distance.

Most of the execution time is needed so search for the nearest neighbor. Fortunately, the nearest neighbor problem is well explored. One of the most efficient approaches is the use of k-d-trees, a space partitioning data structure. K-d-trees use splitting planes parallel to the axis to separate the space into partitions, each containing one point. Consequently, a nearest neighbor search algorithm can address directly the space and doesn't has to calculate the distances to all points. Once a point is found, no point with less distance exists and therefore the algorithm can be terminated.

### C. Fusion of Vision and Laser Data

After the transformation for the image data has been computed and executed, the image data and laser data can be merged into one single colored point cloud. Let  $C_V$  denote the point cloud of the stereo vision system and  $C_L$  the one obtained by the laser range-finders.  $C_V$  and  $C_L$  are aligned properly. To merge  $C_V$  and  $C_L$  into  $C_M$  it is adequate to search for every point  $p_j^L$  in  $C_L$  the point  $p_j^V$  in  $C_V$ , with the smallest distance between the points. The resulting point  $p_j^M$  will have the position of point  $p_j^L$  and the color of point  $p_j^V$ . Furthermore, a standard distribution has been selected for the probability function  $f_j(d_j)$ . Equation 5 has to be calculated and stored for each point in  $C_M$ :

$$f_j(d_j) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{1}{2}\left(\frac{d_j}{\sigma}\right)^2\right) \quad (5)$$

With the distance  $d_j = d(p_j^L, p_j^V)$  between  $p_j^L$  and  $p_j^V$ .  $\sigma$  is the standard deviation and has to be selected in a adequate way.

It depends on the quality of the vision data. The size of  $C_M$  will be equal to the size of  $C_L$ .

## V. EXPERIMENTAL RESULTS

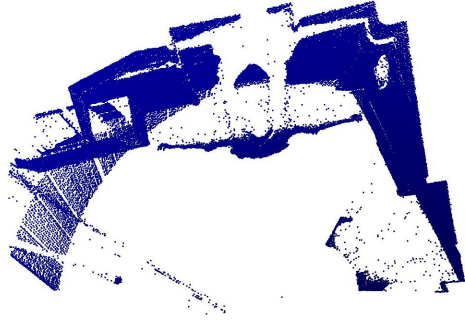


Fig. 5. Scan of left laser range-finder



Fig. 6. Scan of right laser range-finder

The execution time for the genetic ICP algorithm is depending on the size of the point clouds. For a size of  $n_L = n_C = 1000$  it can compute about 3000 generations per second. Figures 5 and 6 show the scan of the left and the right laser



Fig. 7. Colored point cloud from the stereo vision algorithm

range finder. The differences in the field of view of the two scanners can clearly be seen. The colored point cloud obtained

from the vision data is shown in figure 7. Compared to the lasers, the vision data is far less accurate, noisier, and covers a larger area of the environment. The spatial resolution is in the order of magnitude of  $10cm$ , compared to  $0.1cm$  for laser range-finders.



Fig. 8. Fused colored point cloud

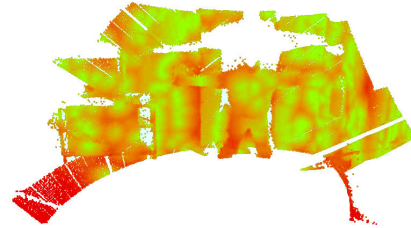


Fig. 9. Probability of the color information

The resulting merged point cloud can be seen in figure 8. Figure 9 shows the probability of the corresponding points. Green areas denote a high probability that the color information is correct, red areas denote a low probability. Figure 10 shows the median error compared to different sizes of the population. The maximal number of generations was set to 10000 generations in the first experiment and to 50000 generations in the second experiment. This experiment was conducted for two different datasets. To compute the median error, our ICP algorithm was executed 20 times for each dataset, each size of population and each maximal number of generations. The generation which produced the best result is depicted in figure 11, again for different sizes of the population and for a maximal number of generations of 10000 and 50000 respectively. As stated out in the figures, a large number of individuals leads to a large number of generations necessary to achieve good results and a too small number of individuals leads to a relatively high medium error. When the number of generations was limited to 50000, the medium error was

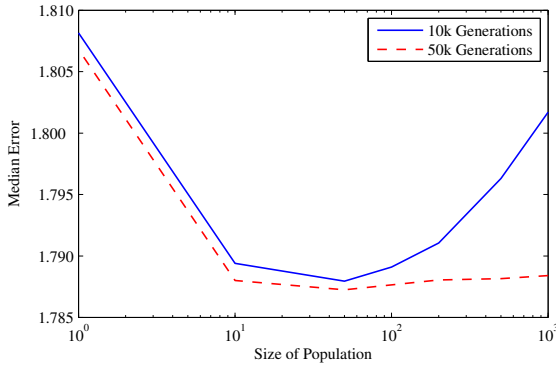


Fig. 10. Correlation between median error and the size of the population

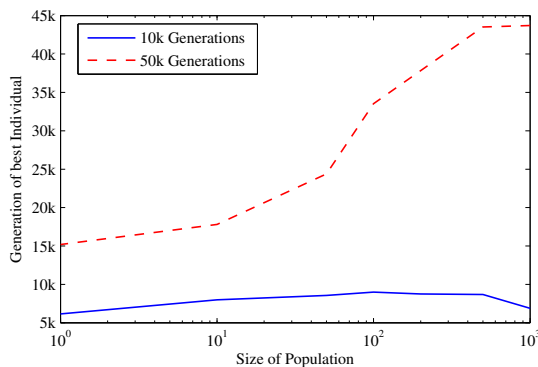


Fig. 11. Correlation between best generation and the size of the population

almost constant and the main differences have occurred in the number of generations needed to compute the best result. Hence, the experiments have shown, that the optimal number of individuals in the population is between 50 and 200. During the experiments with this population size, the best result was found in the first 20000 to 25000 generations, so the maximal number of generations should be set to this order of magnitude.

## VI. CONCLUSION

We presented a genetic ICP algorithm, which is capable of aligning a colored point cloud from a stereo vision system with a point cloud obtained from laser range-finders. These laser range-finders have to be calibrated, to ensure a proper alignment of the point cloud.

In a next step, different scans have to be merged into one big map of the environment and a compact representation for a colored point cloud has to be found. This can be accomplished

by the modification of existing SLAM algorithms. Meanwhile, methods for the recognition and classification of objects in point clouds or other representations of the environment can be developed.

## VII. ACKNOWLEDGMENTS

This work is supported in part within the DFG excellence initiative research cluster *Cognition for Technical Systems – CoTeSys*, see also [www.cotesys.org](http://www.cotesys.org).

## REFERENCES

- [1] P. J. Besl and H. D. McKay. A method for registration of 3-d shapes. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 14(2):239–256, 1992.
- [2] Y. Chen, J. Ni, and S. Wu. Dynamic calibration and compensation of a 3d laser radar scanning system. *Robotics and Automation, 1993. Proceedings., 1993 IEEE International Conference on*, pages 652–658 vol.3, 2-6 May 1993.
- [3] B. K. P. Horn. Extended gaussian images. *Proceedings of the IEEE*, 72:1671–1686, 1984.
- [4] B. K. P. Horn. Closed-form solution of absolute orientation using unit quaternions. *Journal of the Optical Society of America. A*, 4(4):629–642, Apr 1987.
- [5] T. Jost and H. Hugli. A multi-resolution scheme icp algorithm for fast shape registration. pages 540–543, 2002.
- [6] K. Klasing, D. Wollherr, and M. Buss. A clustering method for online segmentation of 3d laser data. In *submitted to IEEE International Conference on Robotics and Automation, 2008*.
- [7] S. Krishnan, P. Y. Lee, J. B. Moore, and S. Venkatasubramanian. Global registration of multiple 3d point sets via optimization-on-a-manifold. In *SGP '05: Proceedings of the third Eurographics symposium on Geometry processing*, page 187, Aire-la-Ville, Switzerland, Switzerland, 2005. Eurographics Association.
- [8] Y. Ma, Z. Wang, M. Bazakos, and W. Au. 3d scene modeling using sensor fusion with laser range finder and image sensor. In *AIPR '05: Proceedings of the 34th Applied Imagery and Pattern Recognition Workshop (AIPR'05)*, pages 224–229, Washington, DC, USA, 2005. IEEE Computer Society.
- [9] A. Makadia, A. I. Patterson, and K. Daniilidis. Fully automatic registration of 3d point clouds. In *CVPR '06: Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1297–1304, Washington, DC, USA, 2006. IEEE Computer Society.
- [10] C. Mei and P. Rives. Calibration between a central catadioptric camera and a laser range finder for robotic applications. *Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference on*, pages 532–537, May 15-19, 2006.
- [11] N. J. Mitra, N. Gelfand, H. Pottmann, , and L. Guibas. Registration of point cloud data from a geometric optimization perspective. In *In R. Scopigno and D. Zorin, editors, Eurographics Symposium on Geometry Processing*, pages 23–32, 2004.
- [12] J. Neira, J. Tardos, J. Horn, and G. Schmidt. Fusing range and intensity images for mobile robot localization, 1999.
- [13] A. Nüchter, H. Surmann, K. Lingemann, and J. Hertzberg. Semantic scene analysis of scanned 3D indoor environments. In *VMV 2003*.
- [14] S. Rusinkiewicz and M. Levoy. Efficient variants of the icp algorithm. In *Proceedings of the Third Intl. Conf. on 3D Digital Imaging and Modeling*, pages 145–152, 2001.
- [15] M. Suppa, S. Kielhofer, J. Langwald, F. Hacker, K. H. Strobl, and G. Hirzinger. The 3d-modeller: A multi-purpose vision platform. In *ICRA*, pages 781–787. IEEE, 2007.
- [16] S. T. and C. B. Fusing imagery and 3d point clouds for reconstructing visible surfaces of urban scenes. *IEEE GRSS/ISPRS Joint Workshop on Remote Sensing and Data Fusion over Urban Areas*, 2007.
- [17] G. Wen, D. Zhu, S. Xia, and Z. Wang. Total least squares fitting of point sets in m-d. In *CGI '05: Proceedings of the Computer Graphics International 2005*, pages 82–86, Washington, DC, USA, 2005. IEEE Computer Society.
- [18] Q. Zhang and R. Pless. Extrinsic calibration of a camera and laser range finder (improves camera calibration). *Intelligent Robots and Systems, 2004. (IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on*, 3:2301–2306 vol.3, 28 Sept.-2 Oct. 2004.