Mobile Netw Appl (2016) 21:313–326
DOI 10.1007/s11036-015-0654-8

# Intelligent Mobile System for Improving Spatial Design Support and Security Inside Buildings

**Janusz Będkowski[1] · Karol Majek[2] · Piotr Majek[2] · Paweł Musialik[2] · Michał Pełka[2] · Andreas Nüchter[3]**

Published online: 19 October 2015
© The Author(s) 2015. This article is published with open access at Springerlink.com

**Abstract** This paper concerns an intelligent mobile application for spatial design support and security domain. Mobility has two aspects in our research: The first one is the usage of mobile robots for 3D mapping of urban areas and for performing some specific tasks. The second mobility aspect is related with a novel Software as a Service system that allows access to robotic functionalities and data over the Ethernet, thus we demonstrate the use of the novel NVIDIA *GRID* technology allowing to virtualize the graphic processing unit. We introduce Complex Shape Histogram, a core component of our artificial intelligence engine, used for classifying 3D point clouds with a Support Vector Machine. We use Complex Shape Histograms also for loop closing detection in the simultaneous localization and mapping algorithm. Our intelligent mobile system is built on top of the Qualitative Spatio-Temporal Representation and Reasoning framework. This framework defines an ontology and a semantic model, which are used for building the intelligent mobile user interfaces. We show experiments demonstrating advantages of our approach. In addition, we test our prototypes in the field after the end-user case studies demonstrating a relevant contribution for future intelligent mobile systems that merge mobile robots with novel data centers.

**Keywords** Intelligent mobile system · 3D object recognition · Qualitative representation and reasoning · 3D mapping

✉ Janusz Będkowski
januszbedkowski@gmail.com

Karol Majek
karolmajek@gmail.com

Piotr Majek
PGMajek@gmail.com

Paweł Musialik
PJMusialik@gmail.com

Michał Pełka
michalpelka@gmail.com

Andreas Nüchter
andreas@nuechti.de

1 Institute of Fundamental Technological Research, Polish Academy of Sciences, ul. Pawińskiego 5B, 02-106 Warsaw, Poland

2 Institute of Mathematical Machines, ul. Krzywickiego 34, 02-078 Warsaw, Poland

3 Julius-Maximilians-University Würzburg, Am Hubland D-97074 Würzburg, Germany

## 1 Introduction

Nowadays, mobile robots are able to efficiently map indoor and outdoor environments. They experience no fatigue and many state of the art systems are robust enough to perform this task autonomously. We focus on an intelligent system capable of 3D mapping and searching objects. Thus, the robot is able to compare current 3D information with the reference model to locate new objects. These objects are considered as objects of potential interest *OPI* that are confirmed by a classification procedure. Accurate 3D models are also used in spatial design support. The spatial design process, in the sense of activity, is described as "Conception → Modeling → Evaluation → Remodeling" cycle in [19]. The work in this paper aims at improving the last three steps. Our goal is to propose a solution to end-users that allows easier improvement and redesigning of already

existing areas, especially those with incomplete or obsolete floor plans. We have built an intelligent mobile system, capable of performing qualitative reasoning.

We propose a modeling → conceptualization → evaluation process based on a 3D scanning and a Qualitative Spatio-Temporal Representation and Reasoning (QSTRR) framework capable of modeling the environment in the sense of generating a qualitative representation. The Qualitative representation used within the context of spatial assistance concept is connected with a framework of multimodal data access for spatial assistance systems [27]. The main goal of this work is to develop an assistance system for supporting designers in decision making based on the Qualitative Spatial Representation and Reasoning (QSRR) [12]. The study of spatial-structural design processes and computational tools such as described in [14] are an example of design support tools in early stages of the design process. As pointed in [13], the main challenge and limitation for QSRR systems is providing an appropriate mathematical tool set allowing for qualitative reasoning without traditional quantitative techniques. Bhatt [4] shows the importance of integrating QSRR techniques with artificial intelligence from the application point of view, such as cognitive robotics or spatial design support. The problem of reasoning about space is discussed in [25]. It simplifies an earlier theory presented in [23] and [24] considered the original approach based on a calculus proposed in [11]. The ontological primitives of the theory include physical objects and regions, an additional set of entities is also defined. The Spatial Assistance System (SAS) described in this paper is defined as a computer implemented decision making system [27] that possesses other analytic abilities that require training, knowledge and expertise. There exist many solutions for spatial perception, but the integration of various spatial knowledge sources for solving complicated problems is an important and complex research problem [2]. The integration problem arises from the fact that conducting the spatial task in a real environment is not bound with one activity but with a set of the mutually dependent tasks, which forces a minimal level of the coordination. Thus, it determines a rich set of the integration problems.

The first step of our spatial design support system consists of acquiring 3D point clouds of the area and merging them into one consistent 3D representation using an improved 6D simultaneous localization and mapping algorithm [20]. During last decade 3D metric information is widely used in inventarization problems [16, 18]. The improvement is related with computational aspects of parallel programming discussed in [7]. In this paper we show current research results of 6D SLAM from the application point of view, thus input data for the qualitative reasoning are based on accurate 3D models of the environment. We achieve better accuracy of final 3D metric map compared to our previous work [3], where we designed ground truth 3D data setup measured with geodetic precision. For spatial design purpose we create a QSTRR model based on this accurate 3D metric map. Finally the evaluation of the area, in connection to the selected spatial design problem, can be done using reasoning capabilities of QSTRR. The preliminary results on QSTRR framework for Spatial Design was shown in [7]. This paper is focused on the integrated intelligent mobile applications into a single mobile system capable of performing qualitative reasoning in the security domain and for spatial design support.

The paper is organized as follows: first we introduce the Qualitative Spatio-Temporal Representation and Reasoning framework, second we describe mobile mapping systems, third we show the search and recognition of 3D objects, fourth we discuss access to robotic functionalities and data with a novel data center using NVIDIA *GRID* technology. Finally, we show experiments convincing end-users about the functionalities of our systems.

## 2 Qualitative spatio-temporal representation and reasoning framework

We designed an ontology for the purpose of encoding spatial design domain knowledge, thus allowing an intelligent mobile robot to perform qualitative reasoning. It allows building a semantic model of an environment using a qualitative spatio-temporal representation. An ontology $O$ is composed of entities: a set of concepts $C$, a set of relations $R$, a set of axioms $A$ (e.g. transitivity, reflexivity, symmetry of relations), a concepts' hierarchy $CH$, a relations' hierarchy $RH$, a set of spatio-temporal events $E_{st}$. It is formulated as the following definition:

$$O = \langle C, R, A, CH, RH, E_{st} \rangle \tag{1}$$

An ontology is a core component of a qualitative reasoning mechanism. A concept is defined as a primitive spatial entity described by a shape $S$ composed of points or polygons in 3D space. We associate a semantic label $SL$ to each shape. Ontology distinguishes two different types of attributes that can be assigned to a concept, quantitative $A_{qn}$ and qualitative $A_{ql}$. We extends the set of four values of qualitative attributes: *real physical object*, *functional space*, *operational space* and *range space* by the new attribute: *empty space*. This new approach helps in reasoning about trajectories in 3D space using state of the art path planning methods. Functional, operational and range spaces are related with spatial artifacts; detailed information are given in [5, 6, 27]. Quantitative attributes are related with physical properties of spatial entities and are as follows: *location*, *mass*, *center of mass*, *moment of inertia* (how much resistance there is to change the orientation about an axis), *material* (friction,

restitution). Therefore, the definition of the concept $C$ is formulated as:

$$C = \langle S, A_{qn}, A_{ql}, SL \rangle. \tag{2}$$

The set of relations $R$ is composed of quantitative and qualitative spatial relations. For topological spatial qualitative relations the Region Connected Calculus (RCC-8) [25] is used. The ontology includes eight different topological relations between two shapes: disconnected $DC$, externally connected $EC$, partial overlap $PO$, equal $EQ$, tangential proper part $TPP$ and its inverse $TPPi$, non-tangential proper part $NTPP$ and its inverse $NTPPi$. Quantitative spatial relations are used for constraining the way entities move relative to another. Our ontology defines following constraints: *origins locked*, *orientations locked*, *origins free*, *orientations free*, *free rotation around one axis sliding*. The ontology provides a mechanism for building world models that assume spatio-temporal relations in different time intervals. It captures changes in 3D space. Our temporal representation takes temporal intervals as a primitive [1]. We distinguish the following qualitative spatio-temporal events $E_{st}$ related with topological spatial relations RCC-8:

- onEnter ($DC \rightarrow EC \rightarrow PO$),
- onLeave ($PO \rightarrow EC \rightarrow DC$),
- onStartInside ($PO \rightarrow TPP \rightarrow NTPP$),
- onStopInside ($NTPP \rightarrow TPP \rightarrow PO$).

These four qualitative spatio-temporal events are used to express the most important spatio-temporal relationships that holds between two concepts in different intervals of time within the context of Spatial Design. In this paper we focus on the two events: onEnter and onLeave. To store the instances of ontology-based elements an instance base $IB^O$ is defined:

$$IB^O = \left\langle I_C^O, I_R^O, I_{E_{st}}^O \right\rangle, \tag{3}$$

where $I_C^O$ contains instances of concepts C, $I_R^O$ contains instances of relations $R$ and $I_{E_{st}}^O$ contains instances of spatio-temporal events. Finally the semantic model is defined as a pair:

$$SM = \left\langle O, IB^O \right\rangle, \tag{4}$$

where $O$ is an ontology and $IB^O$ is an instance base related to ontology O. The ontology is known a-priori but the instance base is being updated during spatial design process. An important tool is the semantic map being defined as a projection of semantic model onto 3D space. All concepts, relations and events have their own user-friendly 3D representation; this is the core of the proposed intelligent mobile user interface.

## 3 Mobile mapping systems

Our mobile mapping systems are shown in Fig. 1. The mobile robots acquire 3D data and register them using improved 6D SLAM algorithm [10]. The lightweight mobile robot PIONEER 3DX is used for 3D mapping of indoor environments and search missions. The Husky robot can provide accurate 3D data with the range of 170 meters in a single 3D scan. We are using this robot to acquire data in the indoor and outdoor environments. We improved the 6D SLAM algorithm by use of semantic classification, loop closing with Complex Shape Histogram (CSH) and parallel implementation. It is shown in Fig. 2, i.e. we have extended the work described in [10]. Important aspects concerning our implementation is given in [8]. Green rectangles correspond to novel algorithmic components using high performance computing in Compute Unified Device Architecture (CUDA). Red rectangles correspond to qualitative reasoning components: semantic classification and semantic 3D scan matching. Semantic classification is used for discrimination of points, thus nearest neighborhood search procedure in semantic 3D scan matching uses this information for finding pairs of closest points having the same semantic label. Semantic labels denote shapes around query points and it is described in Section 4 within the context of computing CSHs. In CSH we use classification of shape round each query point into two classes: *flat*, *not flat*. In semantic 3D scan matching we extend the number of classes by *ceiling* and *floor*. *Ceiling* corresponds to points with label: *flat* above robot with normal vector pointing down, *floor* corresponds to points with label: *flat* below robot with normal vector pointing up. The main part of semantic 3D scan matching is a modified Iterative Closest Points (ICP) algorithm. Our improvements concentrate on a parallel computing implementation and discriminating points

**Fig. 1** Intelligent mobile mapping systems. *Left*: autonomous mobile robot PIONEER 3DX equipped with 3D laser measurement unit. *Right*: mobile mapping system based on robot Husky equipped additionally with accurate geodetic 3D laser Z+F IMAGER 5010

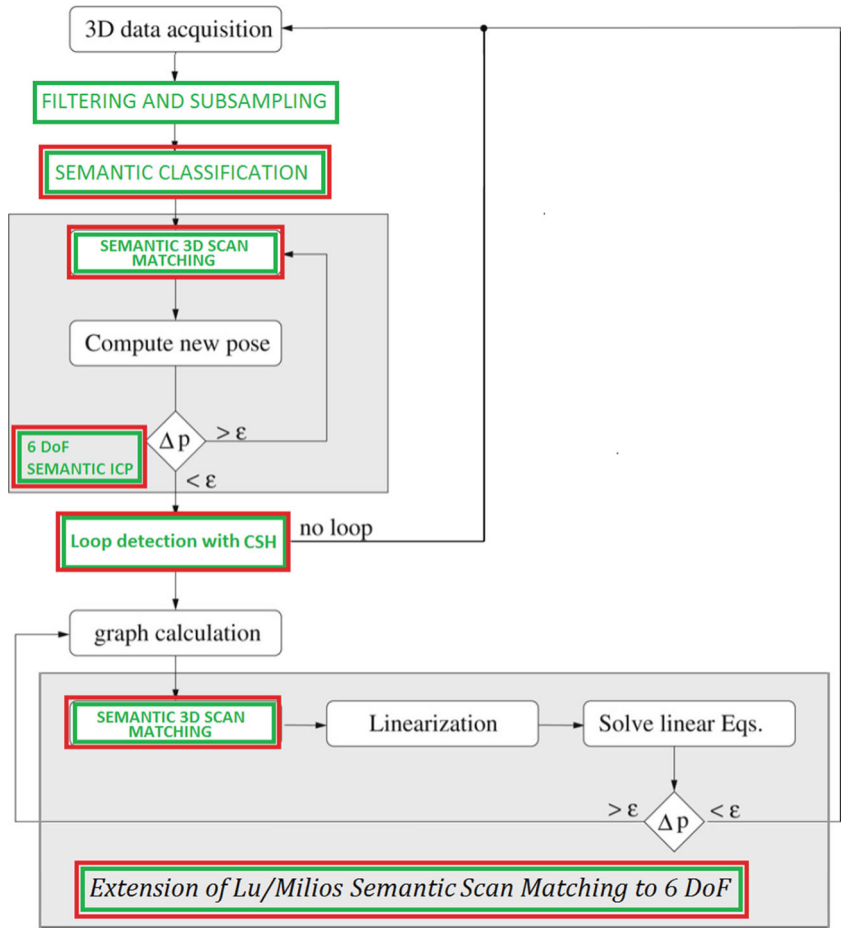


**Fig. 2** Improved 6D SLAM algorithm by semantic classification and parallel implementation. Green rectangles correspond to novel algorithmic components using high performance computing in Compute Unified Device Architecture *CUDA*, red rectangles correspond to qualitative reasoning components: semantic classification and semantic 3D scan matching

into four classes during the nearest neighborhood search procedure. The key concept of the ICP algorithm can be summarized in two steps:

1. Compute correspondences between the two scans (Nearest Neighbor Search),
2. Compute a transformation which minimizes distance between corresponding points.

Iteratively repeating these two steps results in convergence to the desired transformation. Semantic discrimination of these correspondences improves the convergence. Range images are defined as model set $M$ where

$$|M| = N_m \tag{5}$$

and data set $D$ where

$$|D| = N_d \tag{6}$$

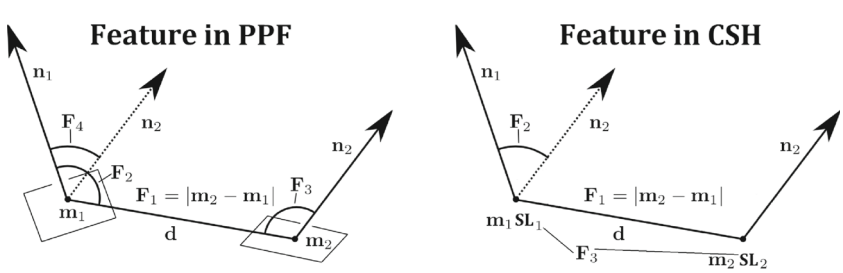


**Fig. 3** Left: Point Pair Feature $\mathbf{F}_{PPF}$ of two oriented points. The component $\mathbf{F}_1$ is set to the distance of the points, $\mathbf{F}_2$ and $\mathbf{F}_3$ to the angle between the normal vectors and the vector defined by the two points, and $\mathbf{F}_4$ to the angle between the two normal vectors. Right: Features $\mathbf{F}_{CSH}$ in CSH. The component $\mathbf{F}_1$ is set to the distance of the points, $\mathbf{F}_2$ to the angle between the two normals, $\mathbf{F}_3$ to combination of semantic labels *SL* denoting shapes around points $\mathbf{m}_1$ and $\mathbf{m}_1$

The goal of semantic 3D scan matching is to minimize following cost function:

$$E(\mathbf{R},\mathbf{t}) = \sum_{i=1}^{N_m}\sum_{j=1}^{N_d} w_{ij} \left\| \mathbf{m}_i^c - \left(\mathbf{R}\mathbf{d}_j^c + \mathbf{t}\right)\right\|^2 \tag{7}$$

where $w_{ij}$ is assigned 1 if the $i^{th}$ point of $M$ correspond to the $j^{th}$ point in $D$ in the sense of minimum distance and the same class. Otherwise $w_{ij}$=0. Class $c$ discriminates points into $flat$, $nonflat$, $ceiling$ or $floor$. $\mathbf{R}$ is the rotation matrix, $\mathbf{t}$ is the translation matrix, $\mathbf{m}_i^c$ corresponds to points of class c from the model set $M$, $\mathbf{d}_j^c$ corresponds to points of class $c$ from the data set $D$. The algorithmic representation of the process is shown in Listing 1. The semantic approach efficiently decreases the time of the 6D SLAM convergence, by requiring a lower number of iterations, especially in the situation where odometry readings are not sufficiently accurate. It is also more reliable than other state of the art approaches for demanding scenarios, such as moving down stairs or mapping harsh environments. We show an example in Section 6 in Fig. 8.

**Algorithm 1** Semantic ICP - parallel computing approach

INPUT: Two point clouds M = $\{m_i\}$, D= $\{d_i\}$, an initial transformation $T_0$, semantic label for each point $\{flat, nonflat, ceiling, floor\}$
OUTPUT: The transformation T, which aligns M and D
$M_{device} \leftarrow M$
$D_{device} \leftarrow D$
$T_{device} \leftarrow T_0$
**for** $iter \leftarrow 0$ to $maxIterations$ **do**
  **for** $i \leftarrow 0$ to $N$ {in parallel} **do**
    $m_i^c \leftarrow$ ClosestPointWithTheSameClass($T_{device} \cdot d_i^c$)
    **if** found $m_i^c$ **then**
      $w_i \leftarrow 1$
    **else**
      $w_i \leftarrow 0$
    **end if**
  **end for**
  $T_{device} \leftarrow \underset{T_{device}}{argmin} \left\{\sum_i w_i \left\| T \cdot d_i^c - m_i^c \right\|^2\right\}$
**end for**
$M \leftarrow M_{device}$
$D \leftarrow D_{device}$
$T \leftarrow T_{device}$

## 4 Recognition of 3D objects

Recognition of 3D objects is composed of *OPI* detection and identification. The pipeline of the autonomous search is shown in Fig. 6. To detect 3D points of *OPI* we use a nearest neighborhood search procedure for each query point in the current 3D measurement. We perform ICP alignment of the current 3D scan to the global reference model. This step is necessary to minimize the number of false detections. Prepared point clouds are processed using Algorithm 2. All of the query points which do not have neighbors within the certain radius are considered as a parts of *OPI*. Points with existing nearest neighbor get the label *non OPI*. In the graphical interface *OPI*s are marked by the red color instead of black. An important issue is the proper initialization of the system by providing a reference 3D model with minimized number of dynamic obstacles.

**Algorithm 2** Detection of *OPI*

INPUT: Reference model $Rm$, Actual measurment $Am$
OUTPUT: Label $\{object_{OPI}, object_{nonOPI}\}$
**for** $i \leftarrow 0$ to $N_{Am}$ **do**
  **if** $NNS(Am_i, Rm)$ **then**
    $label_{Am_i} \leftarrow object_{OPI}$
  **else**
    $label_{Am_i} \leftarrow object_{nonOPI}$
  **end if**
**end for**

To identify 3D objects we build a knowledge base, thus we prepare a training data set composed of objects with assigned semantic labels $SL$. Recognition of 3D objects is performed by classification of the observed object into semantic label based on knowledge base. In this application we have two semantic labels: *OPI* and *not OPI*. We assign these semantic labels manually, thus the advantage of the system is the capability to train for recognition of different objects. To characterize the 3D objects we use the CSH. The CSH (Algorithm 3) is an extension of Point Pair Feature (PPF) [15], which is similar to the surflet-pair feature from [26, 28]. The PPF method uses a point pair feature that describes the relative position and orientation of two oriented points. The feature $\mathbf{F}_{\mathrm{PPF}}$ for two points $\mathbf{m}_1$ and $\mathbf{m}_2$ with normals $\mathbf{n}_1$ and $\mathbf{n}_2$ and distance $\mathbf{d} = \mathbf{m}_2 - \mathbf{m}_1$ is defined as:

$$\mathbf{F}_{PPF}(\mathbf{m}_1,\mathbf{m}_2) = (||\mathbf{d}||_2, \angle(\mathbf{n}_1,\mathbf{d}), \angle(\mathbf{n}_2,\mathbf{d}), \angle(\mathbf{n}_1,\mathbf{n}_2)) \tag{8}$$

where $\angle(\mathbf{a},\mathbf{b}) \in [0;\pi]$ denotes the angle between two vectors. CSH reduces the dimension of the feature space and uses characterisation of shape assigned by semantic label around query point, thus

$$\mathbf{F}_{CSH}(\mathbf{m}_1,\mathbf{m}_2) = (||\mathbf{d}||_2, \angle(\mathbf{n}_1,\mathbf{n}_2), c(\mathbf{m}_1,\mathbf{m}_2)) \tag{9}$$

where $\angle(\mathbf{a},\mathbf{b}) \in [0;\pi]$ denotes the angle between two vectors, $c(\mathbf{a},\mathbf{b}) \in$ (0,1,2,...n) denotes combination of semantic labels $SL$ denoting shapes around query points $\mathbf{a}$ and $\mathbf{b}$. In the current implementation of CSH: 0 corresponds to

(*flat*,*flat*), 1 corresponds to *flat*, *non flat*) or (*non flat*, *non flat*), 2 corresponds to (*non flat*, *non flat*). The classification into flat and not flat labels is performed with Principal Component Analysis (PCA) of the neighborhood for a given query point. Figure 3 depicts the feature in PPF and CSH. In our experiments the Euclidean distance between points is quantized to 20 buckets. The combination of shapes is quantized to 3 buckets and the angle between normal vectors computed for a pair of points is quantized to 5 buckets. The advantage of CSH over PPF is a reduction of feature space, thus assuming the same quantization the histogram of CSH is composed of 300 buckets and histogram of PPF is composed of 2500 buckets. A neglectable disadvantage of CSH is the higher computation complexity over PPF determined by classification of neighborhood for each query point into certain semantic label *SL*, thus a parallel algorithm for fast CSH computation is implemented. In average on a cloud of 15000 3D data points the CSH is computed within 4 seconds on GeForce GTX680 and within 1.7 second using an approximated method.

### 4.1 Intelligent system for 3D objects identification

Object identification is performed with a Support Vector Machine. The SVM is a tool for machine learning performing a task of classification and approximation. SVMs belong to the family called kernel machines with linear or nonlinear kernel. SVM solves the task of classification and regression methods with the optimization constraints. Our system is capable of building SVM models for 3D objects. These models are used for classification purpose, therefore the system is able of performing qualitative reasoning by assigning a semantic label *SL* for observed 3D object. The training and classification phases are shown in Figs. 4 and 5. Input is obtained by a 3D scanner. For such 3D point cloud we compute the CSH. The processed data are stored in database. Each training set is composed of positive examples (CSH of query 3D object) and negative examples (CSH of different 3D objects). Classification is done using the SVM with linear kernel function. The task of searching for the classification of the optimal hyperplane separates the data belonging to the different classes. The training set is defined as $(\mathbf{x}_i, \mathbf{y}_i)_{1 \le i \le N}$ where each example $\mathbf{x}_i \in \mathbf{R}^d$, d being a dimension of an input space, belongs to the class labeled by $\mathbf{y}_i \in \{-1, 1\}$. The goal of supervised learning is to find a hyperplane, which divides the set of examples. Thus, all the points with the same label suppose to be classified to the same category. Before the classification SVM has to be trained to obtain a model describing the query 3D object. The classification is performed by the SVM that responds binary $positive, negative$. $Positive$ label corresponds to potential object of interest *OPI*. Qualitative reasoning is performed by assigning semantic label to the result of the classification (Fig. 6).

**Algorithm 3** CSH

INPUT: Normalized cloud of points: P = $\{p_i(x, y, z) : x, y, z \in < -1, 1 >\}$, normal vectors for points P: N = $\{n_i\}$, semantic labels for points P: $SL = sl_i \in \{flat, notflat\}$ OUTPUT: CSH vector of 300 integer elements: $\{csh_k\}$

```
for k ← 0 to 300 do
  csh_k ← 0
end for
for i ← 0 to numberofpoints do
  for j ← 0 to numberofpoints do
    if i ≠ j then
      if sl_i == flat and sl_j == flat then
        slbucket ← 0
      endif
      if sl_i ≠ sl_j
        slbucket ← 1
      end if
      if sl_i == notflat and sl_j == notflat then
        slbucket ← 2
      end if
      angle ← acos(dotProduct(n_i, n_j))
      anglebucket ← floor(angle/pi × 5.0)
      dist ← ||p_i, p_j||
      dbucket ← floor(dist/(2 × √2) × 20.0)
      index ← slbucket × 100 + anglebucket × 20+
      dbucket
      csh_index ← csh_index + 1
    end if
  end for
end for
```

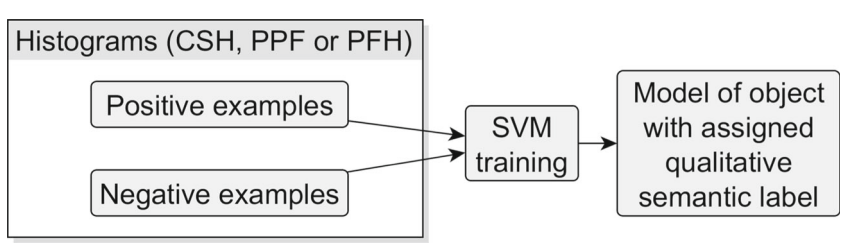


**Fig. 4** System structure for 3D objects recognition - SVM training phase

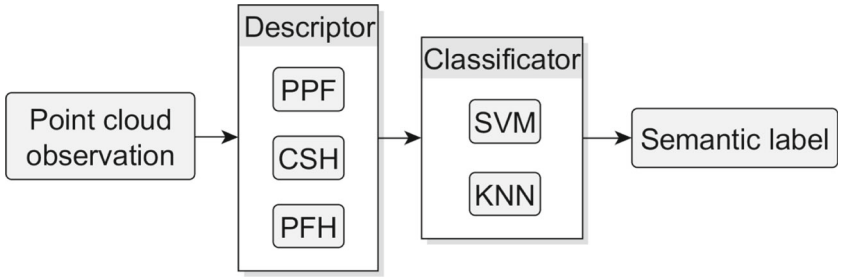


**Fig. 5** System structure for 3D objects recognition - SVM qualitative reasoning phase

## 5 Intelligent mobile user interfaces

The intelligent mobile user interfaces are implemented using a Software as a Service *SaaS* model with the capability to perform High Performance Computing (HPC) in the Cloud. The HPC approach is a relatively new topic, especially in mobile robotics. We are using it for the task of 3D mapping and for detection of *OPI*. HPC with Graphic Processing Unit *GPU* virtualization technology allows many users to have access to computationally demanding applications in the Cloud. We have presented this technology applied for the Search and Rescue ICARUS system in the field [22]. We decided to provide needed computation power through a mobile data center based on an NVIDIA *GRID* server (Supermicro RZ-1240i-NVK2 with two VGX K2 cards - in total 4 GPUs with 4GB RAM each) capable of *GPU* virtualization. In the ICARUS project we have used the Citrix XenApp infrastructure for building the *SaaS* model. In XenApp, many users can share a single *GPU* by the use of Microsoft Windows Server 2012 system or a group of published applications. XenApp model allows sharing a single *GPU* for numerous application using *CUDA*. Published applications can be accessed as SaaS from a web browser after installing the Citrix Receiver - thin client compatible with all popular operating systems for mobile devices (Mac OS X, Linux, Windows, iOS, Android etc.). The limitation is that the newest *GRID* compatible *GPU* drivers support only the *CUDA 5.5* programming framework and the *GRID GPU* processors offer *CUDA 3.0* compute capability. The provided functionalities are sufficient for our application. Besides *CUDA*, *GRID* technology supports building applications with 3D rendering (OpenGL 4.4). Thus, in our system it is used for remote graphical user interface for authorized personnel. Our system is able to improve global awareness of the situation by providing 3D renderings of the mapped environment with marked *OPI*s over Ethernet in real time. Our *SaaS* based security system enables integration with existing closed-circuit television systems, thus it reduces blind spots in existing safety infrastructure. The advantage of the approach is that all information from the robot is transmitted over Ethernet as renders, which is significantly faster than sending raw data. Thus, for example Crisis Management Centers will be able to use robotic information to coordinate crisis action. We tested the experimental setup in Institute of Mathematical Machines shown in Fig. 7. Our system is able provide intelligent mobile user interfaces for 20 users simultaneously. This allows to quick set up command and control centers in search and rescue scenarios.

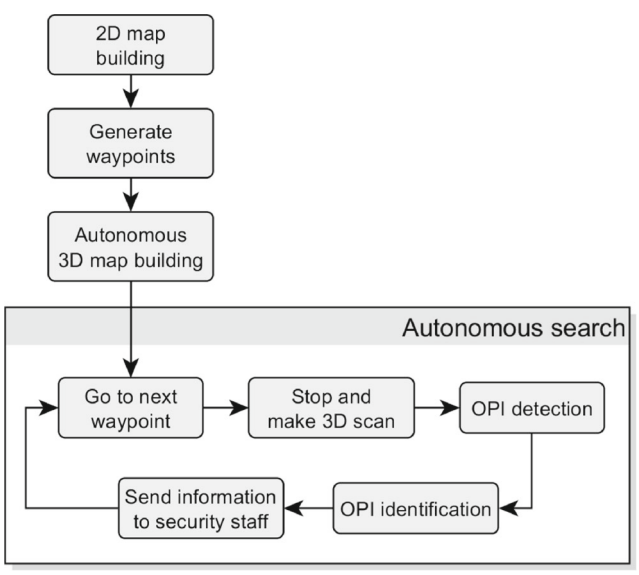


**Fig. 6** The pipeline of the autonomous search

**Fig. 7** Experimental setup of simulated Crisis Management Center in Institute of Mathematical Machines for testing intelligent mobile user interfaces in a SaaS model

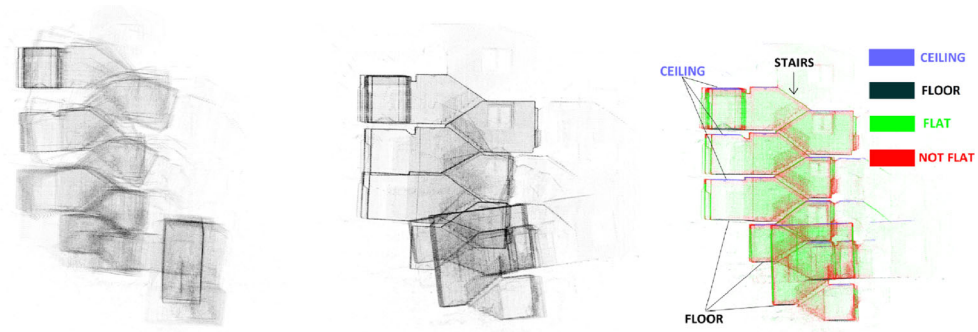


**Fig. 8** Example of a demanding 3D mapping scenario - robot moving down stairs. Left - 3D map based on odometry readings, middle 3D map after state of the art data registration based on Iterative Closest Point, right 3D map obtained with proposed semantic 3D scan matching. We use the classes: *flat*, *not flat*, *ceiling* and *floor* in semantic 3D scan matching

**Fig. 9** Objects in classification experiment comparing Complex Shape Histogram, Point Pair Feature and Point Feature Histogram

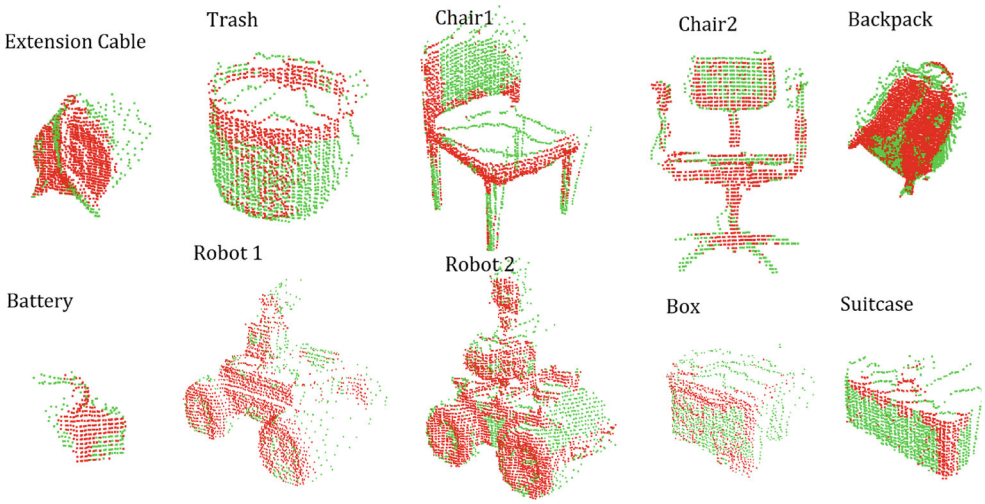


**Fig. 10** 3D point clouds for each object from Fig. 9. Colors correspond to different semantic labels *SL*: red-non flat, green-flat

## 6 Experiments

We performed an end user case study. This shows the need of providing following functionalities:

- accurate 3D mapping,
- search and detecting changes in 3D environment,
- closed-circuit television CCTV camera system modeling and evaluation.

Therefore, we performed the following experiments as a contribution of our research. All figures shows intelligent mobile interfaces available over Ethernet via a novel SaaS system based on the NVIDIA $GRID$ technology.

### 6.1 Accurate 3D mapping

An improved 6D SLAM algorithm as shown in Fig. 2 is capable building accurate 3D models of the environment. Figure 8 shows real task experiment demonstrating the advantage of proposed approach compared to state of the art within the context of demanding scenario - robot moving down the stairs. Our 3D mapping solution is providing more accurate maps in less number of iterations. We observed that map relaxation calculations can be reduced from 100 iterations down to 10 having comparable accuracy, thus the time of computation is drastically reduced. It is observed that semantic discrimination of points improves scan matching, thus objects such as ceiling and floor are not merged.

### 6.2 Mobile intelligent security application – search and detecting changes in 3D environments

The system is capable building 3D model of the environment as a reference data. This data is used for detecting geometric changes in the environment and for classification. Searching objects is performed in autonomous mode in stop-scan fashion. The robot is visiting goals and in each certain location it is performing 3D scan. This scan is compared with reference 3D model of the environment for detecting geometric changes. These geometrical changes are input for the classifier capable distinguish different objects into predefined classes: *object of potential interest* or *other*, provided during supervised learning. The system is informing the operator about the detected changes and the classification result. We compared Complex Shape Histogram, Point Pair Feature and Point Feature Histogram in an experiment where we performed different classification strategies of 10 objects shown in Fig. 9. Each object was scanned 100 times assuming different field of view. Example scans with visualized semantic discrimination of points - different semantic labels $SL$ are depicted in Fig. 10, thus we built representative data set. For each scan we

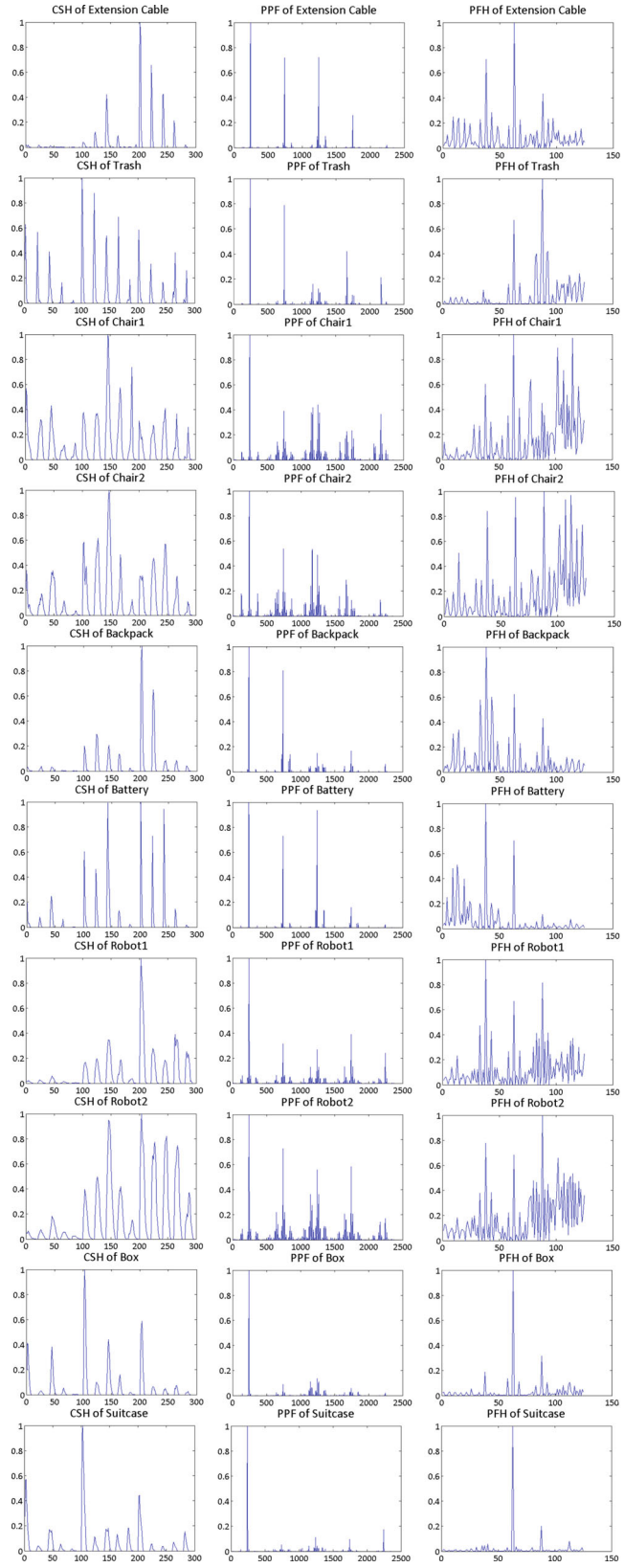


**Fig. 11** Example CSH, PPF and PFH histograms computed for data shown in Fig. 10 obtained by scaning objects from Fig. 9

**Fig. 12** Classification results after training strategy where we assigned positive label (y=1) only to single object and negative label to other objects. $CR$ (correct rate), $CRT$ (correct rate for positive label), $TT$ (training time), $CT$ (classification time) and $SV$ (percentage of support vectors). In last rows we calculated how many times CSH was better than PFH and PPF, thus for example CSH>PFH: 6/10 means that $CR$ of CSH was higher than $CR$ of PFH in 6 of 10 (repeated 100 times) training/classification trials

| | | SVM | | | | | KNN | | |
|---|---|---|---|---|---|---|---|---|---|
| | | CR [%] | CRP [%] | TT [s] | CT [s] | SV [%] | CR [%] | CRP [%] | CT [s] |
| Extension Cable: y=1 All: y=-1 | CSH | 98.99 | 96.36 | 0.0783 | 0.0037 | 6.8 | 99.38 | 97.82 | 0.5000 |
| | PPF | 99.09 | 95.68 | 0.1332 | 0.0416 | 9.5 | 99.56 | 97.32 | 5.1396 |
| | PFH | 95.91 | 80.88 | 0.1472 | 0.0012 | 12.6 | 97.46 | 85.76 | 0.1284 |
| Trash: y=1 All: y=-1 | CSH | 99.37 | 95.28 | 0.0286 | 0.0036 | 4.7 | 99.36 | 96.98 | 0.5438 |
| | PPF | 99.60 | 98.14 | 0.0972 | 0.0383 | 5.2 | 99.64 | 97.28 | 5.3728 |
| | PFH | 99.52 | 95.86 | 0.0177 | 0.0009 | 3.7 | 99.59 | 96.66 | 0.1441 |
| Chair 1: y=1 All: y=-1 | CSH | 98.92 | 94.24 | 0.0360 | 0.0039 | 6.9 | 99.08 | 92.98 | 0.4989 |
| | PPF | 99.48 | 98.36 | 0.1071 | 0.0395 | 8.1 | 99.58 | 97.02 | 5.0936 |
| | PFH | 99.80 | 98.88 | 0.0267 | 0.0010 | 4.8 | 99.49 | 95.78 | 0.1301 |
| Chair 2: y=1 All: y=-1 | CSH | 99.50 | 97.74 | 0.0336 | 0.0038 | 7.1 | 99.26 | 96.92 | 0.4340 |
| | PPF | 99.61 | 96.78 | 0.1011 | 0.0399 | 9.0 | 99.77 | 97.92 | 4.6147 |
| | PFH | 98.99 | 95.32 | 0.0462 | 0.0010 | 6.7 | 99.14 | 95.40 | 0.1141 |
| Backpack: y=1 All: y=-1 | CSH | 99.21 | 96.80 | 0.0352 | 0.0035 | 5.5 | 99.16 | 95.46 | 0.4264 |
| | PPF | 99.23 | 98.00 | 0.1057 | 0.0387 | 7.8 | 99.63 | 99.14 | 4.6200 |
| | PFH | 98.50 | 93.80 | 0.0491 | 0.0010 | 7.7 | 98.32 | 91.02 | 0.1134 |
| Battery: y=1 All: y=-1 | CSH | 99.49 | 97.20 | 0.0272 | 0.0036 | 4.1 | 99.39 | 96.50 | 0.4684 |
| | PPF | 99.22 | 96.16 | 0.1051 | 0.0380 | 7.6 | 99.71 | 98.02 | 4.8822 |
| | PFH | 98.98 | 94.52 | 0.0526 | 0.0010 | 7.9 | 98.42 | 94.96 | 0.1222 |
| Robot 1: y=1 All: y=-1 | CSH | 99.43 | 97.94 | 0.0366 | 0.0038 | 6.7 | 99.09 | 96.48 | 0.4331 |
| | PPF | 99.59 | 99.04 | 0.1076 | 0.0410 | 9.4 | 99.89 | 99.82 | 4.5863 |
| | PFH | 99.63 | 99.28 | 0.0565 | 0.0010 | 7.0 | 99.08 | 95.52 | 0.1123 |
| Robot 2: y=1 All: y=-1 | CSH | 99.63 | 97.94 | 0.0214 | 0.0037 | 4.2 | 99.31 | 96.46 | 0.4355 |
| | PPF | 99.32 | 99.02 | 0.1068 | 0.0411 | 10.3 | 99.79 | 99.74 | 4.6083 |
| | PFH | 99.04 | 96.98 | 0.0727 | 0.0010 | 7.7 | 98.87 | 95.94 | 0.1134 |
| Box: y=1 All: y=-1 | CSH | 99.71 | 98.70 | 0.0270 | 0.0038 | 4.3 | 99.22 | 96.58 | 0.4459 |
| | PPF | 99.85 | 99.82 | 0.1075 | 0.0386 | 5.6 | 99.44 | 97.44 | 4.6914 |
| | PFH | 98.91 | 95.70 | 0.0439 | 0.0011 | 7.3 | 98.21 | 94.74 | 0.1163 |
| Suitcase: y=1 All: y=-1 | CSH | 98.50 | 92.82 | 0.0512 | 0.0039 | 7.4 | 99.02 | 94.44 | 0.4094 |
| | PPF | 98.28 | 94.62 | 0.1218 | 0.0401 | 9.5 | 99.39 | 98.14 | 4.3876 |
| | PFH | 98.53 | 93.54 | 0.0426 | 0.0011 | 7.7 | 98.30 | 89.56 | 0.1079 |
| | CSH>PFH | 6/10 | 6/10 | | | | 8/10 | 9/10 | |
| | CSH>PPF | 3/10 | 3/10 | | | | 0/10 | 1/10 | |

computed CSH, PPF and PFH. Example histograms are shown in Fig. 11. We compared two classifiers SVM and $k$NN, thus we depict results in Figs. 12 and 13. We use two quantitative measures:

1. *Correct Rate*

$$CR = \frac{Correctly\ Classified\ Samples}{Classified\ Samples} \tag{10}$$

**Fig. 13** Classification results after different training strategies where we tested different configurations of assigned labels. $CR$ (correct rate), $CRT$ (correct rate for positive label), $TT$ (training time), $CT$ (classification time) and $SV$ (percentage of support vectors). In last rows we calculated how many times CSH was better than PFH and PPF, thus for example CSH>PFH: 4/4 means that $CR$ of CSH was higher than $CR$ of PFH in 4 of 4 (repeated 100 times) training/classification trials

| | | SVM | | | | | KNN | | |
|---|---|---|---|---|---|---|---|---|---|
| | | CR [%] | CRP[%] | TT [s] | CT [s] | SV[%] | CR[%] | CRP[%] | CT [s] |
| Extension cable:y=-1 Trash:y=-1 Chair[1]:y=-1 Chair[2]:y=-1 **Backpack:y=1** **Battery:y=1** **Robot[1]:y=1** **Robot[2]:y=1** Box:y=-1 **Suitcase:y=1** | CSH | 95.88 | 96.08 | 0.1968 | 0.0047 | 14.5 | 97.78 | 97.91 | 0.4380 |
| | PPF | 95.59 | 96.48 | 0.1942 | 0.0503 | 17.7 | 98.56 | 99.13 | 4.6343 |
| | PFH | 93.93 | 93.90 | 0.2747 | 0.0014 | 17.1 | 95.86 | 96.33 | 0.1146 |
| **Extension cable:y=1** **Trash:y=1** Chair[1]:y=-1 Chair[2]:y=-1 **Backpack:y=1** **Battery:y=1** Robot[1]:y=-1 Robot[2]:y=-1 **Box:y=1** **Suitcase:y=1** | CSH | 99.86 | 99.87 | 0.0169 | 0.0034 | 3.2 | 99.75 | 99.83 | 0.4255 |
| | PPF | 99.98 | 99.98 | 0.0847 | 0.0371 | 4.3 | 99.77 | 99.91 | 4.5784 |
| | PFH | 99.85 | 99.89 | 0.0192 | 0.0009 | 4.5 | 99.47 | 99.59 | 0.1110 |
| Extension cable:y=-1 **Trash:y=1** Chair[1]:y=-1 Chair[2]:y=-1 **Backpack:y=1** **Battery:y=1** Robot[1]:y=-1 Robot[2]:y=-1 Box:y=-1 **Suitcase:y=1** | CSH | 97.94 | 97.36 | 0.1195 | 0.0038 | 10.6 | 98.70 | 98.09 | 0.4160 |
| | PPF | 97.76 | 97.46 | 0.1559 | 0.0401 | 14.4 | 99.09 | 99.08 | 4.4676 |
| | PFH | 96.09 | 94.75 | 0.1449 | 0.0011 | 15.0 | 96.00 | 94.88 | 0.1096 |
| Extension cable:y=-1 Trash:y=-1 Chair[1]:y=-1 Chair[2]:y=-1 Backpack:y=-1 Battery:y=-1 **Robot[1]:y=1** **Robot[2]:y=1** Box:y=-1 Suitcase:y=-1 | CSH | 99.55 | 99.27 | 0.0337 | 0.0039 | 6.4 | 99.56 | 99.39 | 0.4205 |
| | PPF | 99.30 | 99.58 | 0.1041 | 0.0408 | 10.0 | 99.83 | 99.97 | 4.5407 |
| | PFH | 99.18 | 98.99 | 0.0483 | 0.0010 | 6.9 | 99.34 | 99.26 | 0.1088 |
| | CSH>PFH | 4/4 | 3/4 | | | | 4/4 | 4/4 | |
| | CSH>PPF | 3/4 | 0/4 | | | | 0/4 | 0/4 | |

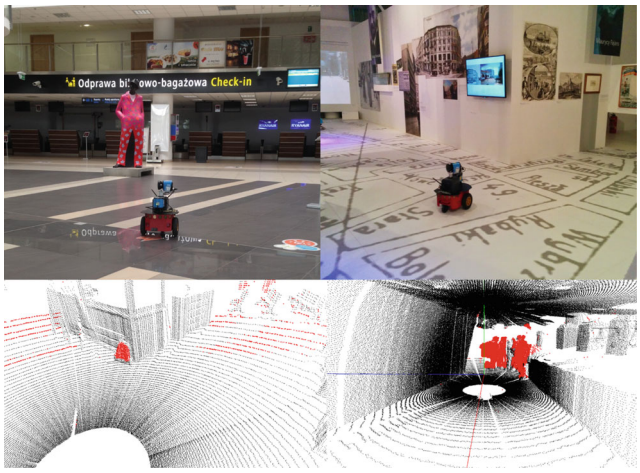


**Fig. 14** The robot tested in the Museum of the History of Polish Jews (Warsaw, Poland) and at the Airport in Łódź, Poland. The results of inspection; Detected changes are marked in red color

**Fig. 15** Underground garage model visualized in WebGL

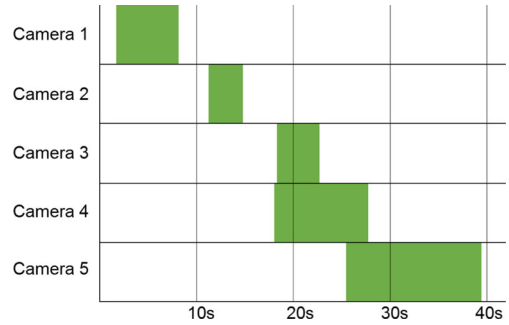


**Fig. 17** Gantt plot for CCTV system evaluation

2. *Correct Rate Positive Samples*

$$CRP = \frac{Correctly\ Classified\ Positive\ Samples}{Classified\ Positive\ Samples} \tag{11}$$

In Fig. 12 we collect classification results after training strategy where we assigned positive label (y=1) only to single object and negative label to other objects. In Fig. 13 we collect classification results after different training strategies where we tested different configurations of assigned labels. We performed 100 cross validation tests for each classifier, thus for each test in training phase we have chosen 50 random samples and for classification we used remaining 50 samples for each object. The mean values of the correct rate $CR$, correct rate for positive label $CRT$, training time $TT$, classification time $CT$ and percentage of support vectors $SV$ are collected in Figs. 12 and 13. In our opinion SVM is more relevant for our application than $k$NN. SVM classifies much faster than $k$NN and needs less information, thus small number of support vectors (typically less than 10 percent of training set) enough for reaching satisfactory correct rate. Another advantage of SVM is the simple implementation of the classification routine on an embedded system limited with computer power. We observed that

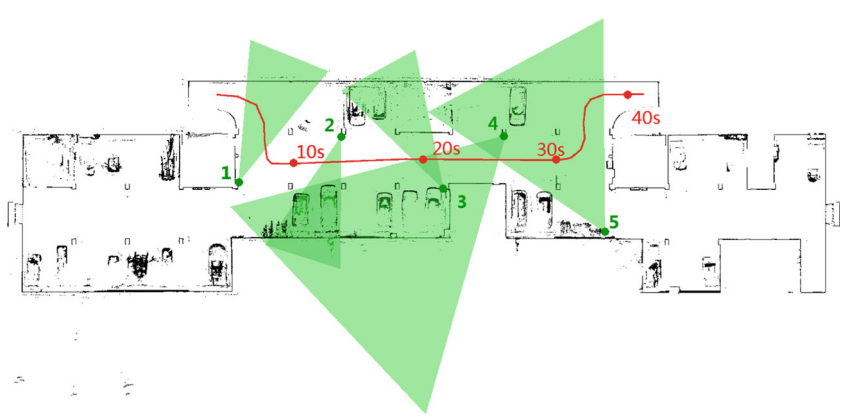


**Fig. 16** Semantic model of the experiment for CCTV system evaluation

**Fig. 18** Simulation of existing CCTV system

CSH performs better than PFH in most cases. PPF gives best classification results but the computational complexity is largest. SVM classifier is faster than $k$NN but correct rate $CR$ in general is slightly lower.

The prototype of the intelligent mobile system was tested in the Museum of the History of Polish Jews and at the Airport in Łódź, Poland, shown in Fig. 14. The system is deployable within 30 minutes. After one hour of 2D mapping we obtained accurate floor plan. Mission planning takes in average two hours. Typical single 3D measurement takes one minute. The robot reaches its next goal within seconds because the average distance between goals should not exceed 5 meters. System was able to detect changes in 3D environment. First *OPI* was classified by our system as backpack, second *OPI* was interpreted by end users as people. This is shown in Fig. 14. Our solution is able to improve the existing security system by eliminating blind spots between installed cameras.

### 6.3 Mobile intelligent spatial design support application

The main goal of the experiment was to validate and improve the surveillance system for the underground garage. The questions from end-users were: does the existing CCTV system is efficient and how to improve the lighting system to increase the awareness of the security staff. As a respond we prepared an experiment in which we built accurate 3D model of underground garage, then we built semantic model of the garage and perform CCTV camera system evaluation using QSTRR framework. To increase the awareness of the security staff we propose the lighting simulation system capable evaluate different light configurations for better CCTV data acquisition. The visualization using WebGL technology is shown in Fig. 15. The staring point is the QSTRR model of the environment with an initial CCTV system. The task is to check the system for blind spots, redesign the system using different capabilities of the framework and validate the solutions. Each camera in the garage was modeled as an entity with a proper range space. The task of the experiment was for a virtual car to drive through the garage, by a predefined path. The assumption was that the car should be visible in at least one camera in every moment of the drive, for the system to pass the validation. The car was considered visible if any of it's part is in the camera range space. The garage was considered empty for the experiment. the result of the experiment is shown in the Gantt plot in Fig. 17 corresponding to semantic model of the experiment shown in Fig. 16. Data are available for download and an immersive visualization is available here.[1] From the Gantt plot we can observe the blind spot between cameras 1,2 and cameras 2,3. This is a very useful quantitative information for end user obtained with proposed qualitative reasoning framework. The subject is closely related to the Art gallery problem. Our system can evaluate existing CCTV system by providing information concerning blind spots (Fig. 17).

The Art gallery problem is a classical problem in geometric optimization [21]. Given a polygonal region, i.e., the art gallery, place as few stationary guards as possible, such that that the entire region is guarded. The Art gallery problem is NP-hard, while the related watchmen problem, where one replaces the stationary guards with a single mobile guard, can be solved efficiently [17, 21]. These geometric optimization focus on theroretical aspects and use simplified models, e.g., all guards have a field of view of 360-degree and unlimited range. Nonetheless, recent progress in art galleries allows solving simplified real-world instances by using an integer programming approach and a finite number of witnesses and a finite number of guard positions [9]. For these algorithms a powerfull integer program (IP) solver such as CPLEX (IBM ILOG CPLEX Optimization Studio) is needed. In contrast to the theoretical grounded work,

[1] http://lider.zms.imm.org.pl/downloads

**Fig. 19** Simulation of improved lighting system of existing CCTV system from Fig. 18

we search for blind spot by the QSTRR model including realistic camera parameter.

The second part of the experiment was the spatial design task related to the improvement of the security system. This experiment shows the possibility of designing the lighting systems in existing buildings using immersive visualization technique. The goal was to increase the awareness of the security personnel. To simulate the light in the model of a building the cameras parameters and locations must be specified. We use 5 cameras from the previous experiment. The lights are described by its positions and the shape which is the same rectangular shape for all lights in this experiment. The lights were simulated using path tracing implemented in NVIDIA Optix library. Several rays are shoot from each pixel and traced. The resulting images have $800 \times 600$ resolution for all cameras. The visualization uses the accurate 3D model, acquired by the mapping system, the camera positions from previous experiment and defined lights. The lights locations are based on the 18 existing lights and the proposed change adds 20 more lights and the presented method shows the result. The results for the 5 cameras and two lighting system layouts are presented in Figs. 18 and 19. Obtained images show that there are dark spots especially in camera 2 and 5. The images generated with more light sources are brighter and they can be more useful for security personnel. The implemented framework allows designers to test the proposed light and CCTV systems before any physical change. Realistic rendering provides reliable information for end users.

## 7 Conclusion

In this paper we have shown the advantages of using the QSTRR framework in security and spatial design applications. The intelligent mobile system is designed for protection of urban environments including critical infrastructures. The method for searching of objects of potential interest was discussed and the results in real task scenarios were presented. The prototype of the system was tested in the Museum of the History of Polish Jews in Warsaw and Airport in Łódź, Poland. The system was able to detect changes in 3D environment and is able to improve the existing security system. Our spatial design support system, based on modeling-validation-remodeling philosophy, allows for creation of accurate model of given environment, modify it using specialized tools and validating the results with simulation. We generated accurate 3D metric models and by adding QSTRR layer, we modeled spatio-temporal events and relations in the environment, which allows for simulating not only the view of the area but also explore the qualitative functionalities of different configurations of spatial entities. The presented experiments show the proof of concept of the system functionality looking from end-user cases point of view. Use of parallel computation allows for increase in quality of the results and lower the computation time to reasonable levels. Our intelligent mobile interfaces implemented using the SaaS model allow accessing data over Ethernet, thus end-users can work simultaneously from different locations.

## 8 Future work

The proposed classification method uses 3D information of objects, thus we will investigate how 2D information from sensor such as camera could improve the performance of the system. There is no limitation for using 2D and 3D information since we can represent the objects using the certain feature space, thus in future work we will investigate the methods for the characterization of 2D objects. The research goal will be the classification system capable working with

2D and 3D information. The new system will be dedicated for cases and applications where user can only provide a 2D images as input.

**Acknowledgments** This work is done with the support of NCN (Polish National Center of Science) project: Methodology of semantic models building based on mobile robot observations Nr: DEC-2011/03/D/ST6/03175. Furthermore, this work is done also with the support of NCBiR (Polish National Center for Research and Development) project: Research of Mobile Spatial Assistance System Nr: LIDER/036/659/L-4/12/NCBR/2013. In addition, the authors received funding from the German Academic Exchange Service DAAD (grant ID 57155328 – Sensor data processing using General Purpose Computations on GPUs")

**Open Access** This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (http://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

## References

1. Allen JF (1983) Maintaining knowledge about temporal intervals. Commun ACM 26:832–843
2. Barkowsky T, Bateman JA, Freksa C, Burgard W, Knauff M (2005) Transregional collaborative research center SFB/TR 8 spatial cognition: reasoning, action, interaction (sonderforschungsbereich/transregio SFB/TR 8 raumkognition: Schließen, handeln, interagieren). it - Information Technology 47(3):163–171
3. Będkowski J, Majek K, Musialik P, Adamek A, Andrzejewski D, Czekaj D (2014) Towards terrestrial 3D data registration improved by parallel programming and evaluated with geodetic precision. Autom Constr 47:78–91
4. Bhatt M (2014) Robotics: concepts, methodologies, tools, and applications, chapter reasoning about space, actions and change: A paradigm for applications of spatial reasoning, pages 315–349 IGI Global
5. Bhatt M, Dylla F, Hois J (2009) Spatio-terminological inference for the design of ambient environments. In: Proceedings of the 9th international conference on Spatial information theory, COSIT'09,. Springer, Berlin, pp 371–391
6. Bhatt M, Hois J, Kutz O, Dylla F (2010) Modelling functional requirements in spatial design. In ER:464–470
7. Będkowski J (2013) Intelligent mobile assistant for spatial design support. Autom Constr 32(0):177–186
8. Będkowski J, Majek K, Nüchter A (2013) General purpose computing on graphics processing units for robotic applications. JOSER 4(1):23–33
9. Borrmann D, de Rezende PJ, de Souza CC, Fekete SP, Friedrich S, Kröller A, Nüchter A, Schmidt C, Tozoni DC (2013) Point guards and point clouds: solving general art gallery problems. In: Proceedings of the 20th ACM Annual Symposium on Computational Geometry (SoCG '13), pp 347–348, Rio de Janeiro, Brazil
10. Borrmann D, Elseberg J, Lingemann K, Nüchter A, Hertzberg J (2008) Globally consistent 3d mapping with scan matching. Robot Auton Syst 56(2):130–142
11. Bowman LC (1981) A calculus of individuals based on connection. Notre Dame Journal Formal Logic 22(3):204–218
12. Cohn AG, Hazarika SM (2001) Qualitative spatial representation and reasoning: an overview. Fundamenta Informaticae 46(1–2):1–29
13. Cohn AG, Renz J (2007) Qualitative spatial representation and reasoning. In: Frank van Harmelen, Vladimir Lifschitz, Bruce Porter (eds) Handbook of Knowledge Representation. Elsevier, New York, pp 551–584
14. Delgado JD, Hofmeyer H (2013) Research engine: A tool to simulate and study spatial-structural design processes. In: Zhang J, Sun C (eds) Global Design and Local Materialization, volume 369 of Communications in Computer and Information Science. Springer, Berlin, pp 96–108
15. Drost B, Ulrich M, Navab N, Ilic S (2010) Model globally, match locally: Efficient and robust 3D object recognition. In: IEEE Conference on CVPR, pp 998–1005
16. Erdős G, Nakano T, Váncza J (2014) Adapting cad models of complex engineering objects to measured point cloud data. CIRP Ann Manuf Technol 63(1):157–160
17. Fekete SP, Klein R, Nüchter A (2006) Online searching with an autonomous robot. Comput Geom Theory Appl (CGTA) 34(2):102–115
18. Lalonde JF, Vandapel N, Hebert M (2006) Automatic three-dimensional point cloud processing for forest inventory Technical Report CMU-RI-TR-06-21. Robotics Institute, Pittsburgh
19. Freksa C, Bhatt M (2012) Analytical computing for spatial design: An artificial intelligence perspective. In: First Annual Conference on Advances in Cognitive Systems, 2012 - Poster Collection. Palo Alto California, pp 35–52
20. Nüchter A, Lingemann K, Hertzberg J, Surmann H (2007) 6D SLAM-3D mapping outdoor environments. J Field Rob 24(8–9):699–722
21. O'Rourke J (1987) Art gallery theorems and algorithms oxford university press
22. Pełka M, Będkowski J, Majek K, Musialik P, Masłowski A, Coelho A, Baptista R, Goncalves R, De Cubber G, Balta H, Sanchez J, Govindraj S (2014) Training and Support system in the Cloud for improving the situational awareness in Search and Rescue (SAR) operations. IEEE International Symposium on Safety, Security, and Rescue Robotics
23. Randell DA, Cohn AG (1989) Modelling topological and metrical properties in physical processes. In: 1st International Conference on the Principles of Knowledge Representation and Reasoning, pp 55–66
24. Randell DA, Cohn AG (1992) Exploiting lattices in a theory of space and time. Computers and Mathematics with Applications 23:459–476
25. Randell DA, Cui Z, Cohn AG (1992) A spatial logic based on regions and connection. In: Proceedings of the 3rd International Conference on Principles of Knowledge Representation and Reasoning (KR'92). Cambridge, MA, October 25–29, pp 165–176
26. Rusu RB, Blodow N, Beetz M (2009) Fast Point Feature Histograms (FPFH) for 3D Registration. In: Proceedings of the IEEE ICRA. Kobe, Japan, pp 3212–3217
27. Schultz C, Bhatt M (2010) A multi-modal data access framework for spatial assistance systems: use-cases with the building information model (bim/ifc). In ISA:39–46
28. Wahl E, Hillenbrand U, Hirzinger G (2003) Surflet-Pair-Relation Histograms: A Statistical 3D-Shape Representation for Rapid Classification. 3DIM, pp 474–482. IEEE Computer Society