

Globally Optimal Inlier Set Maximization for Atlanta Frame Estimation

Kyungdon Joo¹ Tae-Hyun Oh² In So Kweon¹ Jean-Charles Bazin³
¹RCV Lab., KAIST ²MIT CSAIL ³CML, KAIST
 {kdjoo369, thoh.mit.edu}@gmail.com {iskweon, bazinjc}@kaist.ac.kr

Abstract

In this work, we describe man-made structures via an appropriate structure assumption, called Atlanta world, which contains a vertical direction (typically the gravity direction) and a set of horizontal directions orthogonal to the vertical direction. Contrary to the commonly used Manhattan world assumption, the horizontal directions in Atlanta world are not necessarily orthogonal to each other. While Atlanta world permits to encompass a wider range of scenes, this makes the solution space larger and the problem more challenging. Given a set of inputs, such as lines in a calibrated image or surface normals, we propose the first globally optimal method of inlier set maximization for Atlanta direction estimation. We define a novel search space for Atlanta world, as well as its parameterization, and solve this challenging problem by a branch-and-bound framework. Experimental results with synthetic and real-world datasets have successfully confirmed the validity of our approach.

1. Introduction

Man-made structures, such as buildings and room layouts, consist of a set of parallel and orthogonal planes or lines. These structures are commonly approximated by the Manhattan world (MW) assumption [7], which is defined by three orthogonal directions. By virtue of its orthogonality and simplicity, MW assumption has been exploited in many vision applications [1, 5, 3, 8, 18]. On the other hand, MW assumption is not verified for a wide range of man-made structures whose horizontal directions are not orthogonal to each other (Fig. 1), for example, non-orthogonal walls [31].

To alleviate this limitation, Schindler et al. [26] proposed *Atlanta world* (AW): the horizontal directions are orthogonal to the vertical (typically gravity) direction, but contrary

*K. Joo and I. S. Kweon were supported by the Technology Innovation Program (No. 2017-10069072) funded by the Ministry of Trade, Industry & Energy, Korea. J.-C. Bazin was partly supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (Ministry of Science, ICT & Future Planning) (No. 01180339). Part of this work was done when T.-H. Oh was in KAIST, and he was partly funded by HBKU QCRI at MIT.

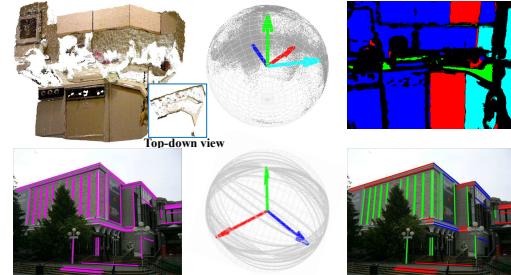


Figure 1: Atlanta frame estimation from surface normals (top) and line normals in a calibrated image (bottom). *Left:* Examples of 3D point clouds (defining surface normals) from NYUv2 dataset [28] and lines from York urban dataset [8]. *Middle:* Distribution of the input normals and the estimated Atlanta directions on the sphere. *Right:* Visualization of the clustered surface normals (segmentation) and lines with respect to the estimated Atlanta directions.

to MW [7], these horizontal directions do not have to be orthogonal to each other. Therefore AW permits to handle a wider range of scenes. However, AW has been surprisingly explored only little [26, 1] and requires specific methods. In addition, MW allows taking advantage of the full orthogonality constraint to greatly reduce the search space. This constraint does not exist in AW, which makes the problem more challenging.

In this paper, we want to estimate the Atlanta world directions. We consider two kinds of input data: surface normals (e.g., from RGBD cameras and laser scanners) and line normals in intrinsically calibrated images, as shown in Fig. 1. To estimate the Atlanta world directions, we need to know which normal verifies the AW (inlier/outlier detection) and which normal belongs to which Atlanta world direction (clustering). To handle outliers and estimate the clustering, we follow the popular strategy of inlier set maximization [14], which aims to maximize the number of inliers (the number of clustered normals) and estimate the underlying model (the Atlanta directions). Several methods have been proposed for inlier set maximization, but none of them is fully satisfactory because they might return a local optimum or are not designed for AW. For example, RANSAC [9] is the most popular method of this strategy but is non-deterministic and does not guarantee to return the optimal solution (in terms of inlier set cardinality).

In contrast to existing methods, our key contribution is the first globally optimal approach for inlier set maximization in AW. Concretely, given a set of input normals, our approach can (1) determine the normal clustering (i.e. find which normal belongs to which unknown-but-sought Atlanta direction), (2) estimate these associated Atlanta directions verifying the AW structure and (3) identify the outlier normals (i.e. which normals do not belong to any of these Atlanta directions), in such a way that (4) the inlier set of clustered normals is maximized. Our approach runs on a spherical representation of the normals and applies a branch-and-bound framework over the definition space of Atlanta world directions on the sphere (Fig. 1). It can be applied to a large range of surface normals (from RGBD cameras, laser scanners, etc.) and lines (images such as perspective, central catadioptric, fisheye and omnidirectional).

2. Related Work

Structure assumptions Various structure assumptions have been studied in the literature [7, 31, 26, 30] (see detailed review in [30]). The Manhattan world (MW) assumption [7] is defined by three orthogonal directions and is widely used [1, 5, 3, 8, 18, 19]. However, the MW assumption is not verified for a wide range of man-made structures such as non-orthogonal walls. Straub et al. proposed a mixture of Manhattan frames (MMF) [31], which consists of multiples MWs. While their method provides a more flexible representation, it cannot enforce constraints between the MWs, such as sharing a common vertical direction. Schindler et al. proposed the Atlanta world assumption (AW) [26], where the horizontal directions are orthogonal to the vertical direction but not necessarily orthogonal to each other. AW can be seen as a relaxed version of MW, or a case of MMF where the multiple MWs share a common vertical direction. In this work, we follow the AW assumption because it can encompass a wide range of man-made structures, such as buildings with different orientations and/or with non-orthogonal walls, while constraining a common vertical direction.

Lines and vanishing points As a structure representation in the 2D image space, vanishing point (VP) estimation has been actively researched [27, 8, 25, 17, 16]. Without structure assumptions, the VPs can be estimated independently [33, 35, 23, 21]. Since information about the scene structure can increase the accuracy, most VP estimation methods rely on the structure assumption, especially the MW assumption [7, 20, 10, 34]. Among them, the branch-and-bound based VP estimation methods [5, 6], which guarantee a globally optimal solution, are closely related to our work. However, they are specifically designed only for MW assumption and thus cannot be applied for AW. Under AW assumption, Schindler et al. [26] estimate VPs by EM method and Lee et al. [22] by a greedy approach for a highly

non-linear cost function. Therefore their results depend on the initialization and might not be the global optimum.

Surface normals Similar to the 2D image domain, the MW assumption has been commonly exploited for surface normals in 3D space to represent man-made structure [28, 12, 11, 18]. There are works [32, 36, 12] that simplify the structural assumption with a dominant ground plane normal, i.e., gravity direction, for efficient structure understanding. Straub et al. [31] introduce Manhattan frame (MF) which jointly captures the directions of the MW in the space of surface normal, and extend to a mixture of MF. Ghanem et al. [11] propose a robust and a non-convex MF estimation utilizing the inherent sparsity of data distribution. Straub et al. [29] propose a real-time MF estimation. Recently, Joo et al. [18] propose globally optimal MF estimation in real-time using a branch-and-bound framework. However, all these methods are sub-optimal and/or limited to MW assumption. In contrast, the proposed approach is globally optimal and is applicable for Atlanta worlds.

3. Problem Statement

3.1. Atlanta Frame

Under AW assumption [26], the horizontal directions are orthogonal to the vertical (typically gravity) direction and do not have to be orthogonal to each other. Therefore we can represent an AW structure as a set of unit direction vectors $\mathcal{V} = \{\mathbf{v}_v, \mathbf{v}_{h_1}, \dots, \mathbf{v}_{h_M}\} = \{\mathbf{v}_j\}_{j=1}^{M+1}$ that consists of a vertical vector $\mathbf{v}_v = \mathbf{v}_1$ (without lack of generality) and a set of M horizontal vectors $\mathbf{v}_{h_m} = \mathbf{v}_{m+1}$, where $\mathbf{v}_v \perp \mathbf{v}_{h_m}$ (AW constraint) for all $m = 1, \dots, M$ where M is the number of horizontal vectors. In the rest of this paper, we call *Atlanta frame* (AF), or *Atlanta directions*, this direction set \mathcal{V} .

3.2. Mathematical Formulation

Our input normals can be a set of surface normals or a set of lines, where a line is represented as the normal of the associated great circle in spherical representation [3] (see Fig. 1). Thus, for simplicity, we can consider the input of our problem as a normal set $\mathcal{N} = \{\mathbf{n}_i\}_{i=1}^N$, where $\mathbf{n}_i \in \mathbb{R}^{3 \times 1}$ is a unit normal and N is the number of input normals.

Our proposed approach follows the popular strategy of inlier set maximization. Given the set of input normals \mathcal{N} , the aim is to maximize the number of clustered normals with respect to the unknown-but-sought Atlanta directions \mathcal{V} . Let \mathcal{S} represent the set of normal-Atlanta direction pairs: $\mathcal{S} = \{(\mathbf{n}_i, \mathbf{v}_j), i=1 \dots N, j=1 \dots M+1\}$ and the set \mathcal{S} be partitioned into an inlier set $\mathcal{S}_I \subseteq \mathcal{S}$ and an outlier set $\mathcal{S}_O \subseteq \mathcal{S}$ with $\mathcal{S}_O = \mathcal{S} - \mathcal{S}_I$. To distinguish inliers/outliers, we follow the popular “residual tolerance method” [9]. Concretely, we consider that the normal-Atlanta direction pair $(\mathbf{n}_i, \mathbf{v}_j)$ is an inlier if their geometric (geodesic) distance is lower than a residual tolerance τ , i.e. $d_n(\mathbf{n}_i, \mathbf{v}_j) = |\angle(\mathbf{n}_i, \mathbf{v}_j)| < \tau$ for a surface normal and

$d_l(\mathbf{n}_i, \mathbf{v}_j) = |\angle(\mathbf{n}_i, \mathbf{v}_j) - \pi/2| < \tau$ for a line normal, where $\angle(\cdot, \cdot)$ is the angle distance between two unit vectors. For simplicity, we will represent both distances as $d(\mathbf{n}_i, \mathbf{v}_j)$ in the paper. This τ is the inlier threshold and can be set easily (e.g. 3°). This inlier pair is noted $(\mathbf{n}_i, \mathbf{v}_j) \in \mathcal{S}_I$ and \mathbf{n}_i is clustered to \mathbf{v}_j . The outlier set \mathcal{S}_O contains the normals that do not belong to any Atlanta directions. The problem can be formulated as an inlier set maximization:

$$\max_{\mathcal{V}} \quad \text{card}(\mathcal{S}_I) \quad (1a)$$

$$\text{s.t.} \quad d(\mathbf{n}_i, \mathbf{v}_j) < \tau, \forall (i, j) \in \mathcal{S}_I \subseteq \mathcal{S} \quad (1b)$$

$$\text{and } \|\mathbf{v}_j\|=1, \forall j \text{ and } \mathbf{v}_1^\top \mathbf{v}_k=0, \forall k = 2 \dots M+1, \quad (1c)$$

where $\text{card}(\cdot)$ is the cardinality of a set. This formulation (1) aims to find the largest inlier set (i.e. maximize the number of clustered inliers) (Eq. (1a)) under an unknown Atlanta frame \mathcal{V} , given an inlier threshold τ on the geometric distance (Eq. (1b)), and such that the AF verify the AW constraint (Eq. (1c)). Solving system (1) provides not only the AF maximizing the number of clustered inliers but also the inlier/outlier identification and the clustering information (i.e. which normal belongs to which direction). System (1) is challenging to solve due to the quadratic constraints (orthogonality and unit norm) and non-linearities (for the geometric distance).

4. Atlanta Frame Parameterization

4.1. Parametrization

The optimization variables of system (1) are the Atlanta directions \mathcal{V} . A naive parametrization is to work directly on the Cartesian coordinates of these directions (i.e. x, y, z). However this complicates the optimization drastically because the $M+1$ directions would be encoded by $3(M+1)$ real values, and it would also require to deal with $M+1$ unit length constraints plus M orthogonality constraints.

To overcome these limitations, we propose an alternative efficient parametrization of the Atlanta frame. Our main idea (see Fig. 2) is based on the fact that, first, all the horizontal directions lie on a great circle (called horizon) whose normal is the vertical direction \mathbf{v}_v ; and second, these horizontal directions can be represented by “walking” a certain distance from a starting location along this horizon. To define the horizon and this starting location, we use a rotation: by applying a rotation \mathbf{R} to the axis $\mathbf{e}_1=(1, 0, 0)$ and $\mathbf{e}_2=(0, 1, 0)$, we obtain two vectors $\mathbf{r}_1=\mathbf{R}\mathbf{e}_1$ and $\mathbf{r}_2=\mathbf{R}\mathbf{e}_2$ of unit length and orthogonal. Without lack of generality, we set $\mathbf{v}_v=\mathbf{r}_1$ and $\mathbf{v}_{h_1}=\mathbf{r}_2$, and we consider \mathbf{v}_{h_1} as the starting location. We write \mathbf{R}_α the rotation around the vertical direction \mathbf{v}_v by the angle α . Then each additional horizontal direction can be defined by $\mathbf{v}_{h_m}=\mathbf{R}_{\alpha_m}\mathbf{v}_{h_1}$ where $2 \leq m \leq M$. We note the angle set $\{\alpha\} = \{\alpha_2, \dots, \alpha_M\}$.

In terms of parametrization, the rotation $\mathbf{R} \in SO(3)$, which defines \mathbf{v}_v and \mathbf{v}_{h_1} , has 3 DoF and can be encoded

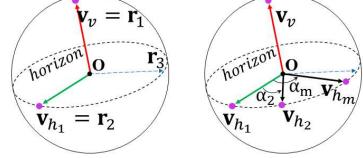


Figure 2: Parametrization of Atlanta frame \mathcal{V} . *Left*: a vertical direction \mathbf{v}_v and first horizontal direction \mathbf{v}_{h_1} are defined by a rotation \mathbf{R} . *Right*: each additional horizontal directions \mathbf{v}_{h_m} ($m \geq 2$) can be obtained by rotating \mathbf{v}_{h_1} around \mathbf{v}_v by an angle α_m .

by 3 parameters by the angle-axis representation. The additional directions $\{\mathbf{v}_{h_m}\}_{m=2}^M$ have 1 DoF rotation angle α_m each. Compared to the naive parametrization, ours has thus only $3+(M-1)=M+2$ parameters and just one constraint (i.e. $\mathbf{R} \in SO(3)$).

4.2. Reformulation

Using the proposed parameterization, system (1) can now be reformulated in an equivalent manner as:

$$\arg \max_{\mathbf{R} \in SO(3), \{\alpha\}} \quad \text{card}(\mathcal{S}_I) \quad (2a)$$

$$\text{s.t. } d(\mathbf{n}_i, \mathbf{v}_j) < \tau, \forall (i, j) \in \mathcal{S}_I \subseteq \mathcal{S}. \quad (2b)$$

Thanks to our parametrization, the reformulation (2) involves fewer parameters and allows to avoid the multiple explicit AW constraints of the original formulation (1c). Unfortunately, system (2) is intractable to solve directly by numerical optimization (known to be NP-hard) [5, 24].

Following the approach of Bazin et al. [4], we introduce the auxiliary binary variable y^{ij} to indicate whether the i -th normal is an inlier to the j -th Atlanta direction (with $y^{ij}=1$) or an outlier (with $y^{ij}=0$). We note $\{y\}$ the set of all the y^{ij} . We can now rewrite system (2) as an equivalent mixed-integer programming problem:

$$\arg \max_{\{y\}, \mathbf{R} \in SO(3), \{\alpha\}} \quad \sum_{i=1}^N \sum_{j=1}^{M+1} y^{ij} \quad (3a)$$

$$\text{s.t. } y^{ij} d(\mathbf{n}_i, \mathbf{v}_j) \leq y^{ij} \tau, \forall i, j \quad (3b)$$

$$y^{ij} \in \{0, 1\}, \forall i, j, \quad (3c)$$

where $\mathbf{v}_1 = \mathbf{v}_v = \mathbf{R}\mathbf{e}_1$, $\mathbf{v}_2 = \mathbf{v}_{h_1} = \mathbf{R}\mathbf{e}_2$, and $\mathbf{v}_{m+1} = \mathbf{v}_{h_m} = \mathbf{R}_{\alpha_m}\mathbf{v}_{h_1} = \mathbf{R}_{\alpha_m}\mathbf{R}\mathbf{e}_2$, as defined in Secs. 3.1 and 4.1. While system (3) is a challenging non-convex problem, we will show in Sec. 5 how to optimize it in a globally optimal way by a branch-and-bound framework.

4.3. Search Space

In our parametrization (Sec. 4.1), the AF search space is composed of first, a *rotation search space* (i.e. $\mathbf{R} \in SO(3)$) and second, an *angle search space* (i.e. the set of 1D angles $\{\alpha\} = \{\alpha_2, \dots, \alpha_M\}$). For the rotation search space, we employ the angle-axis form to represent a rotation \mathbf{R} , that is a three-dimensional vector \mathbf{r} in a ball B_π of radius π , whose direction $\mathbf{r}/\|\mathbf{r}\|$ and norm $\|\mathbf{r}\|$ respectively specify the axis and angle of the rotation \mathbf{R} [13]. In the angle-axis form, any rotation can be represented by a point in the ball B_π . As

for the angle space, we use 1D angle space L_π with ranges $(-\pi, \pi]$. Due to the symmetry of the problem (antipodal points indicate the same direction), we consider just half of the rotation ball B_π and limit the range of 1D angle space to $(-\pi/2, \pi/2]$.

5. Branch-and-Bound

Branch-and-bound (noted BnB) is a general framework for global optimization [15]. The basic idea of BnB is to divide the search space into smaller sub-spaces (i.e. branching) and remove the spaces that cannot contain a solution better than the current one. This removing decision is made by a feasibility test and the associated bounds. Iteratively, the size of the sub-spaces decreases and the estimated solution converges to the optimal solution.

5.1. Branching Operations

As discussed in Sec. 4.3, our search space consists of a rotation search space by angle-axis form and a set of angle search spaces. Following the convention of the angle-axis form [13], let D_π be an initial cube of half-length π that tightly encloses the ball B_π . The cube representation makes the BnB subdivision operation simple because it is axis-aligned. We divide the rotation search space into smaller congruent sub-spaces by octal subdivision of the cube for branching. Each rotation sub-space can be defined as:

$$\mathcal{C}_r(\bar{\mathbf{r}}, \sigma) = \{\mathbf{x} \in \mathbb{R}^3 \mid (\mathbf{x} - \bar{\mathbf{r}}) \in [-\sigma, \sigma]^3\}, \quad (4)$$

where $\bar{\mathbf{r}}$ is the center of the cube (i.e. in angle-axis parametrization) and σ is the half side length of the cube. In 1D angle space L_π , the angle search space is bi-divided to angle sub-space and can be defined as:

$$\mathcal{C}_a(\bar{\alpha}, \sigma) = \{x \in \mathbb{R} \mid (x - \bar{\alpha}) \in [-\sigma, \sigma]\}, \quad (5)$$

where $\bar{\alpha}$ is the center of the angle sub-space and σ is the half side length of the 1D space.

Then, an Atlanta search cube can be represented as $\mathcal{C}(\sigma) = \mathcal{C}_r(\bar{\mathbf{r}}, \sigma) \times \mathcal{C}_a(\bar{\alpha}_2, \sigma) \times \cdots \times \mathcal{C}_a(\bar{\alpha}_M, \sigma)$. Any point in \mathcal{C} corresponds to specific values for angle-axis \mathbf{r} and for the angles $\{\alpha_m\}$, and thus uniquely defines an AF. Without loss of generality, we assume in the following that each of the parameter σ in an Atlanta search cube \mathcal{C} have the same value for rotation and angle spaces. For writing simplification, we write $\mathcal{C} = \mathcal{C}_r(\bar{\mathbf{r}}) \times \mathcal{C}_a(\bar{\alpha}_2) \times \cdots \times \mathcal{C}_a(\bar{\alpha}_M)$.

5.2. Bounding Operations

The BnB feasibility test aims to compute a lower and upper bounds of the optimal value of the objective function that can be obtained in a given search cube. In the following, we explain how to compute the bounds for our problem.

5.2.1 Computation of bounds

We want to answer the question: how to compute the lower and upper bounds of the optimal number of inlier clus-

tered normals that can be obtained in a given Atlanta search cube \mathcal{C} . The answer is: the lower and upper bounds are the solutions of the below systems (8) and (9) respectively. In these systems, $\bar{\mathcal{V}}$ is the (known) AF corresponding to the center of the cube $\mathcal{C} = \mathcal{C}_r(\bar{\mathbf{r}}) \times \mathcal{C}_a(\bar{\alpha}_2) \times \cdots \times \mathcal{C}_a(\bar{\alpha}_M)$, i.e. by using $\bar{\mathbf{r}}, \bar{\alpha}_2 \cdots \bar{\alpha}_M$, we have

$$\bar{\mathcal{V}} = \{\bar{\mathbf{v}}_j\}_{j=1}^{M+1} = \{\bar{\mathbf{v}}_v, \bar{\mathbf{v}}_{h_1}, \bar{\mathbf{v}}_{h_2}, \dots, \bar{\mathbf{v}}_{h_M}\} \quad (6)$$

$$= \{\bar{\mathbf{R}}\mathbf{e}_1, \bar{\mathbf{R}}\mathbf{e}_2, \bar{\mathbf{R}}_{\alpha_2}\bar{\mathbf{R}}\mathbf{e}_2, \dots, \bar{\mathbf{R}}_{\alpha_M}\bar{\mathbf{R}}\mathbf{e}_2\}, \quad (7)$$

where $\bar{\mathbf{R}} = \mathbf{R}_{\bar{\mathbf{r}}}$ corresponds to the rotation at the center of $\mathcal{C}_r(\bar{\mathbf{r}})$ and $\bar{\mathbf{R}}_{\alpha_m} = \mathbf{R}_{\bar{\alpha}_m}$ is the rotation around the vertical direction $\bar{\mathbf{v}}_v = \bar{\mathbf{R}}\mathbf{e}_1$ by the angle $\bar{\alpha}_m$ at the center of $\mathcal{C}_a(\bar{\alpha}_m)$.

The system for the lower bound is:

$$\max_{\{y\}} \sum_{i=1}^N \sum_{j=1}^{M+1} y^{ij} \quad (8a)$$

$$\text{s.t. } y^{ij} d(\mathbf{n}_i, \bar{\mathbf{v}}_j) \leq y^{ij} \tau, \forall i, j \quad (8b)$$

$$y^{ij} \in \{0, 1\}, \forall i, j \quad (8c)$$

The system for the upper bound is:

$$\max_{\{y\}} \sum_{i=1}^N \sum_{j=1}^{M+1} y^{ij} \quad (9a)$$

$$\text{s.t. } y^{ij} d(\mathbf{n}_i, \bar{\mathbf{v}}_j) \leq y^{ij} (\tau + \sqrt{3}\sigma), \forall i, j = 1, 2 \quad (9b)$$

$$y^{ij} d(\mathbf{n}_i, \bar{\mathbf{v}}_j) \leq y^{ij} (\tau + (1 + \sqrt{3})\sigma), \forall i, j = 3 \cdots M+1 \quad (9c)$$

$$y^{ij} \in \{0, 1\}, \forall i, j, \quad (9d)$$

where Eq. (9b) is for $\bar{\mathbf{v}}_1 = \bar{\mathbf{v}}_v$ and $\bar{\mathbf{v}}_2 = \bar{\mathbf{v}}_{h_1}$ (depending only on $\mathcal{C}_r(\bar{\mathbf{r}})$), while Eq. (9c) is for $\bar{\mathbf{v}}_{m+1} = \bar{\mathbf{v}}_{h_m}$ with $2 \leq m \leq M$ (depending on $\mathcal{C}_r(\bar{\mathbf{r}})$ and $\mathcal{C}_a(\bar{\alpha}_m)$).

Proof. Proofs for systems (8) and (9) are in Sec. 5.2.2. \square

The unknowns of the systems (8) and (9) are just the binary y^{ij} (note that the Atlanta frame $\bar{\mathcal{V}}$ is known here, from the given cube center). While the values of y^{ij} could be obtained by integer programming, they can be obtained by simply checking whether a normal verifies the inlier distance constraint or not. Therefore given an Atlanta search cube \mathcal{C} , we now have a method to compute the lower and upper bounds of the optimal number of inlier clustered normals that can be obtained by any AF contained in \mathcal{C} . These bounds can then be used in a branch-and-bound framework.

5.2.2 Proof of bounds

In this section, we prove that systems (8) and (9) provide valid lower and upper bounds in the current cube \mathcal{C} , that is to say that the optimal inlier cardinality c^* that can be obtained in \mathcal{C} is between these lower l and upper u bounds, i.e. $l \leq c^* \leq u$. Here the values l and u are the sum of y^{ij} , i.e. the objective value of the two systems. For writing simplification, let us note $\mathcal{V}^* = \{\mathbf{R}^*\mathbf{e}_1, \mathbf{R}^*\mathbf{e}_2, \mathbf{R}_{\alpha_2}^*\mathbf{R}^*\mathbf{e}_2, \dots, \mathbf{R}_{\alpha_M}^*\mathbf{R}^*\mathbf{e}_2\}$ the AF of \mathcal{C} providing the optimal number of c^* inliers.

Proof of the lower bound Since c^* is the optimal cardinality with an inlier threshold τ among all $\mathcal{V} \in \mathcal{C}$ and since

$\bar{\mathcal{V}}$ (the cube center) is a particular element of \mathcal{C} , then it is obvious that the cardinality l obtained by $\bar{\mathcal{V}}$ with the same inlier threshold is such that $l \leq c^*$.

Proof of the upper bound This proof is more challenging. First, we write a rotation as \mathbf{R}_r in matrix form and as \mathbf{r} in its associated angle-axis form (in \mathbb{R}^3). To bound the uncertainty angle by rotation, we recall the following Lemma.

Lemma 1. [13] For any vector \mathbf{u} and two rotations, represented as $\mathbf{R}_{\mathbf{r}_1}$ and $\mathbf{R}_{\mathbf{r}_2}$ in matrix form and as \mathbf{r}_1 and \mathbf{r}_2 in angle-axis form,

$$\angle(\mathbf{R}_{\mathbf{r}_1}\mathbf{u}, \mathbf{R}_{\mathbf{r}_2}\mathbf{u}) \leq \|\mathbf{r}_1 - \mathbf{r}_2\|. \quad (10)$$

Based on Lemma 1, we can now derive uncertainty angles for our rotation sub-space $\mathcal{C}_r(\bar{\mathbf{r}})$ and angle sub-space $\mathcal{C}_a(\bar{\alpha})$.

Lemma 2 (Uncertainty angle of rotation space). Given a rotation cube $\mathcal{C}_r(\bar{\mathbf{r}}, \sigma_r)$ of half side length σ_r centered at $\bar{\mathbf{r}}$, then for any vector \mathbf{u} and $\forall \mathbf{r} \in \mathcal{C}_r(\bar{\mathbf{r}}, \sigma_r)$,

$$\angle(\mathbf{R}_{\mathbf{r}}\mathbf{u}, \mathbf{R}_{\bar{\mathbf{r}}}\mathbf{u}) \leq \sqrt{3}\sigma_r. \quad (11)$$

Proof. Inequality (11) can be derived as follows:

$$\angle(\mathbf{R}_{\mathbf{r}}\mathbf{u}, \mathbf{R}_{\bar{\mathbf{r}}}\mathbf{u}) \leq \|\mathbf{r} - \bar{\mathbf{r}}\| \quad (\text{by Lemma 1}) \quad (12)$$

$$\leq \sqrt{3}\sigma_r, \quad (13)$$

where Eq. (13) follows from $\max_{\mathbf{r}} \|\mathbf{r} - \bar{\mathbf{r}}\| = \sqrt{3}\sigma_r$ (the half space diagonal of the rotation cube) for $\mathbf{r} \in \mathcal{C}_r(\bar{\mathbf{r}}, \sigma_r)$. \square

Lemma 3 (Uncertainty angle of 1D angle space). Given an angular cube $\mathcal{C}_a(\bar{\alpha}, \sigma_\alpha)$ of half side length σ_α centered at $\bar{\alpha}$, then for any vector \mathbf{u} and $\forall \alpha \in \mathcal{C}_a(\bar{\alpha}, \sigma_\alpha)$,

$$\angle(\mathbf{R}_\alpha\mathbf{u}, \mathbf{R}_{\bar{\alpha}}\mathbf{u}) \leq \sigma_\alpha, \quad (14)$$

where \mathbf{R}_α and $\mathbf{R}_{\bar{\alpha}}$ are calculated from angle-axis forms which share a same rotation axis (e.g., vertical direction \mathbf{v}_v) but different rotation angles α and $\bar{\alpha}$.

Proof. Inequality (14) can be derived as follows:

$$\angle(\mathbf{R}_\alpha\mathbf{u}, \mathbf{R}_{\bar{\alpha}}\mathbf{u}) \leq \|\alpha - \bar{\alpha}\| \quad (15)$$

$$\leq \sigma_\alpha, \quad (16)$$

since \mathbf{R}_α and $\mathbf{R}_{\bar{\alpha}}$ have the same rotation axis but different rotation angles α and $\bar{\alpha}$, the Euclidean distance between their angle-axis forms is $\|\alpha - \bar{\alpha}\|$, and by Lemma 1, Eq. (15) is satisfied. Eq. (16) follows from $\max_\alpha \|\alpha - \bar{\alpha}\| = \sigma_\alpha$ (the half space of 1D angle space) for $\alpha \in \mathcal{C}_a(\bar{\alpha}, \sigma_\alpha)$. \square

For the proof of the upper bound, we prove the first constraint (Eq. (9b)) in system (9), i.e. uncertainty angle for $\mathbf{v}_1 = \mathbf{v}_v = \mathbf{R}\mathbf{e}_1$ and $\mathbf{v}_2 = \mathbf{v}_{h_1} = \mathbf{R}\mathbf{e}_2$ (depending only on \mathbf{R} , not α). Let $(\mathbf{n}_i, \mathbf{R}^*\mathbf{e}_j)$ be an inlier match, i.e. $\angle(\mathbf{n}_i, \mathbf{R}^*\mathbf{e}_j) \leq \tau$. Then, by triangle inequality

$$\angle(\mathbf{n}_i, \bar{\mathbf{R}}\mathbf{e}_j) \leq \angle(\mathbf{n}_i, \mathbf{R}^*\mathbf{e}_j) + \angle(\bar{\mathbf{R}}\mathbf{e}_j, \mathbf{R}^*\mathbf{e}_j) \quad (17)$$

$$\leq \tau + \sqrt{3}\sigma. \quad (\text{by Lemma 2}) \quad (18)$$

We now prove the second constraint (Eq. (9c)) in system (9), i.e. uncertainty angle for the additional horizontal

directions $\mathbf{v}_{m+1} = \mathbf{v}_{h_m} = \mathbf{R}_{\alpha_m}\mathbf{v}_{h_1} = \mathbf{R}_{\alpha_m}\mathbf{R}\mathbf{e}_2$. Similarly to the first constraint, we can write by triangle inequality

$$\angle(\mathbf{n}_i, \bar{\mathbf{R}}_{\alpha_m}\bar{\mathbf{R}}\mathbf{e}_2) \quad (19)$$

$$\leq \angle(\mathbf{n}_i, \mathbf{R}_{\alpha_m}^*\mathbf{R}^*\mathbf{e}_2) + \angle(\bar{\mathbf{R}}_{\alpha_m}\bar{\mathbf{R}}\mathbf{e}_2, \mathbf{R}_{\alpha_m}^*\mathbf{R}^*\mathbf{e}_2) \quad (20)$$

$$\leq \angle(\mathbf{n}_i, \mathbf{R}_{\alpha_m}^*\mathbf{R}^*\mathbf{e}_2) + \angle(\bar{\mathbf{R}}_{\alpha_m}\bar{\mathbf{R}}\mathbf{e}_2, \bar{\mathbf{R}}_{\alpha_m}\mathbf{R}^*\mathbf{e}_2) \quad (21)$$

$$+ \angle(\mathbf{R}_{\alpha_m}^*\mathbf{R}^*\mathbf{e}_2, \bar{\mathbf{R}}_{\alpha_m}\mathbf{R}^*\mathbf{e}_2) \quad (22)$$

$$\leq \tau + \sqrt{3}\sigma_r + \sigma_{\alpha_m} \quad (\text{by Lemmas 2 and 3}) \quad (22)$$

$$= \tau + (1 + \sqrt{3})\sigma. \quad (\sigma = \sigma_{\mathbf{r}} = \sigma_{\alpha_m}) \quad (23)$$

These two constraints mean that if a match is an inlier by the Atlanta frame $\mathcal{V}^* \in \mathcal{C}$ w.r.t. the residual threshold τ , then it is also an inlier by $\bar{\mathcal{V}}$ (the center of \mathcal{C}) w.r.t. the threshold $\tau + \sqrt{3}\sigma$ for $(\bar{\mathbf{v}}_v, \bar{\mathbf{v}}_{h_1})$ and $\tau + (1 + \sqrt{3})\sigma$ for $\bar{\mathbf{v}}_{h_m}$ (with $m \geq 2$). Therefore if \mathcal{V}^* provides c^* inliers, then system (9) provides $u \geq c^*$ inliers, which concludes the proof of the upper bound.

Finally, we have proved the validity of our lower and upper bounds. Throughout the BnB iterations, the cube size σ decreases (by cube subdivision). Therefore the gap between the lower and upper bounds decreases, i.e. the lower and upper bounds converge, and in turn BnB converges to the globally optimal solution since the bounds are valid. This concludes the proof. \square

5.3. Search Procedure

Our BnB procedure is formalized in Alg. 1 using a breadth-first search (depth-first search is also applicable). The cube list \mathcal{L} is initialized with the bounded search space of the parameters (Sec. 4). At each BnB iteration, each cube of \mathcal{L} is subdivided into 2^{M+2} sub-cubes of half size along each dimension and the associated bounds are computed. Then we remove from the list all the cubes whose upper bound is lower than the maximum lower bound l^* obtained so far because it means that, even in the best case, it is sure that these cubes cannot obtain an inlier cardinality higher than l^* . The procedure stops when at least one cube whose lower bound equals the maximum upper bound is obtained or when it reaches a desired accuracy. Finally the AF corresponding to the center of that cube is returned, i.e. this AF globally maximizes the number of inliers.

5.4. Efficient Bound Computation

Most of the BnB execution time is spent on verifying the inlier constraints in systems (8) and (9), for the lower and upper bound computations. For the line case, the number of input normals is typically only up to a few hundreds (i.e. number of lines extracted in an image), which is computationally inexpensive. In contrast, the number of surface normals can be orders of magnitude higher: for example, a VGA depth map provides around 300,000 surface normals, and laser scanners in the order of millions, which will drastically increase the execution time.

Algorithm 1 BnB for inlier set maximization in Atlanta world

Initialize the cube list \mathcal{L} with the bounded search space (Sec. 4)
repeat
 Subdivision ($\sigma \leftarrow \sigma/2$) of each cube C_i of \mathcal{L}
for each cube C_i of \mathcal{L} **do**
 Get the Atlanta frame $\bar{\mathcal{V}}$ of the cube center (Eq. (6))
 Compute the lower l_i and upper u_i bounds (Sec. 5.2)
end for
 $l^* = \max_i l_i, i^* = \arg \max_i u_i, u^* = u_{i^*}, \mathcal{V}^* = \mathcal{V}_{C_{i^*}}$
 Remove all the cubes from \mathcal{L} such that $u_i < l^*$
until $\exists i$ such that $l_i = u^*$ (i.e. at least one cube whose lower-bound is u^*) or it reaches a desired accuracy level.
Return: \mathcal{V}^* (i.e. the AF maximizing the number of inliers)

To speed up the bound computation for large-scale surface normals, we apply the relaxation method of Joo et al. [18], which achieves a constant time bound computation with respect to the number of input normals. Their method was originally developed in the context of Manhattan frame, but we can adapt it for Atlanta frame using our proposed bound derivations (Sec. 5.2.2) and our Atlanta frame parametrization (Sec. 4.1). Due to space limitation, the formal definition of our relaxation is available in the supplementary material. In the context of surface normals, the basic idea of the relaxation is to relax the circular inlier region (geodesic distance up to τ on the sphere) by a bounded rectangular region along azimuth and elevation axes. It allows us to leverage the integral image technique on a 2D domain with azimuth and elevation axes rather than 3D sphere, i.e. to compute the bounds in constant time.

Relationship with original problem Let the distance $d_{AF}(\mathcal{V}, \mathcal{V}')$ denote a distance metric between two Atlanta frames \mathcal{V} and \mathcal{V}' , and $\mathbb{S}=\{\mathcal{V}^*\}$ denote the set of globally optimal solutions obtained by solving the problem like system (3). We write H_d the Hausdorff distance which measures the distance between two sets and is defined over a distance d . Then, we have the following relationship.

Lemma 4. *Given any fixed input measurements, let $\mathbb{S}_o(\tau)$ and $\mathbb{S}_r(\tau)$ respectively be the sets of the globally optimal solutions obtained by solving system (3) and its relaxed version, with the inlier threshold τ . For $\epsilon > 0$, suppose there exists $\mathbb{S}_o(\tau^\#)$ such that $H_{d_{AF}}(\mathbb{S}_r(\tau), \mathbb{S}_o(\tau^\#)) \leq \epsilon$, then*

$$\forall V \in \mathbb{S}_r(\tau), \left| d_{AF}(V, \mathbb{S}_o(\tau)) - H_{d_{AF}}(\mathbb{S}_o(\tau), \mathbb{S}_o(\tau^\#)) \right| \leq \epsilon. \quad (24)$$

Formal definitions and the detailed proof can be found in the supplementary material. Lemma 4 elucidates that, regardless of input data, the relaxation gap between the original and relaxed problems can be approximated alternately by the gap of the solutions $\mathbb{S}_o(\tau)$ and $\mathbb{S}_o(\tau^\#)$, if $\mathbb{S}_o(\tau^\#)$ represents the relaxed problem close enough (ϵ -gap). This provides an alternative way to represent the relaxation gap by the gap between the same class of two problems, i.e.

$H_{d_{AF}}(\mathbb{S}_o(\tau), \mathbb{S}_o(\tau^\#))$. For instance, the original problem has a circular inlier region R_τ^o for a direction \mathbf{v} on a sphere, and suppose its relaxed inlier region R_τ^r forms a rectangular shape circumscribing the circle. Then, we can derive another original problem that has a circular inlier region $R_{\tau^\#}^o$ but circumscribing R_τ^r ; $\text{Area}(R_\tau^o) \subset \text{Area}(R_\tau^r) \subset \text{Area}(R_{\tau^\#}^o)$. If the data distribution is “simple” enough such that $\mathbb{S}_r(\tau)$ is close to $\mathbb{S}_o(\tau^\#)$, then instead of measuring the relaxation gap, we can deduce the gap from the difference between $\mathbb{S}_o(\tau)$ and $\mathbb{S}_o(\tau^\#)$. If the data distribution is “complicated” such that solutions of the problems are sensitively changed according to even a small change of inlier threshold, then ϵ may be increased; it loosens the tightness of Eq. (24), i.e. ϵ is data dependent.

6. Experimental Results

In this section, we present experimental results for synthetic and real-world datasets. Our algorithm has been implemented in MATLAB and runs on a laptop equipped with an Intel i7-4790K 4.0GHz CPU and 32GB RAM. Given the inlier threshold τ in degrees and the number of targeted (horizontal) Atlanta directions M set by the user, our algorithm runs in a fully automatic manner.

6.1. Synthetic Data

We now study the behavior of our algorithm, especially in terms of *optimality* and *convergence*. For this, we randomly generated N input normals (both lines and surface normals) in AW containing $M + 1$ Atlanta directions. We applied a Gaussian noise of 3° to the orientations of the normals to mimic data noise. We also corrupted a certain percentage p of the input data to create outliers, i.e. normals that do not belong to any Atlanta directions. Unless specifically mentioned, we set $M = 2$ (i.e. 3 directions in total), $N = 100$ normals, $\tau = 3^\circ$ and $p = 20\%$ of outliers.

RANSAC techniques For comparison purpose, we implemented some RANSAC techniques: 4-line RANSAC for lines and 2-normal RANSAC for surface normals.¹ For 4-line RANSAC, at each iteration we randomly select four lines. We hypothesize the two horizontal directions \mathbf{v}_{h_1} and \mathbf{v}_{h_2} by intersecting two lines each, and then compute the vertical direction \mathbf{v}_v by cross-product of \mathbf{v}_{h_1} and \mathbf{v}_{h_2} . As for 2-normal RANSAC, we randomly select two surface normals and use them as hypothesis for the horizontal directions \mathbf{v}_{h_1} and \mathbf{v}_{h_2} . The vertical directions \mathbf{v}_v is then computed by cross-product of these two horizontal directions.

Procedure We compared our method to the above RANSAC techniques, and the quality of the results is measured by the number of inliers. The number of RANSAC iter-

¹The described RANSAC techniques are designed for 3 Atlanta directions ($M = 2$), but their process is easily generalizable to different number of Atlanta directions, as detailed in the supplementary material.

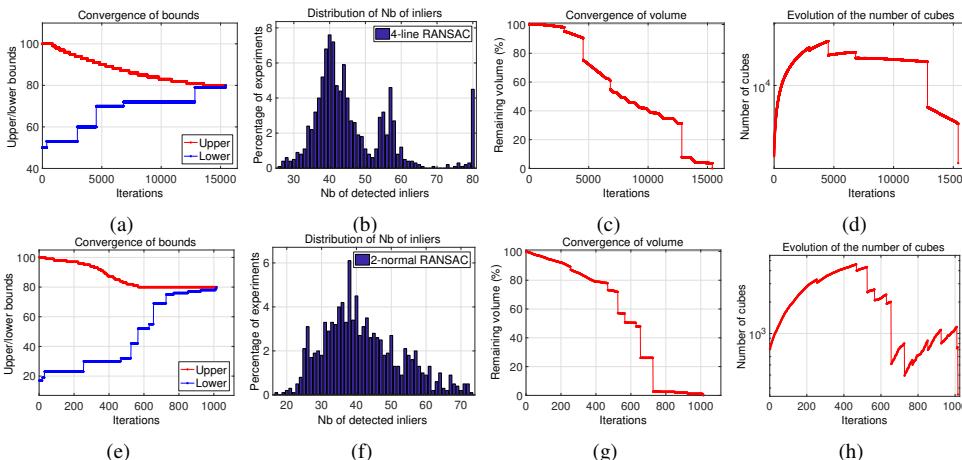


Figure 3: Representative results obtained for synthetic data on lines (top row) and surface normals (bottom row). (a, e): Convergence of the lower and upper bounds of the number of inliers. (b, f): Distribution of the number of inliers detected by RANSAC (4-line and 2-normal RANSAC) on the same problem instance. (c, g): Convergence of the volume of the BnB search space. (d, h): Evolution of the number of BnB cubes.

ations is automatically computed with the true outlier ratio, a guaranteed accuracy of 99% and the minimal sampling (i.e. 4lines or 2 normals) [14]. As RANSAC is not deterministic, we repeated the experiments 1,000 times with the same input data to obtain the inlier cardinality distribution.

Analysis Fig. 3 shows representative results for synthetic data on lines and surface normals. Figs. 3(a) and 3(e) show that the lower and upper bounds converge for both line and surface normal, which illustrate the convergence of the proposed algorithm and prove the bounds are valid. For comparison, Figs. 3(b) and 3(f) show the distribution of the number of inliers obtained by RANSAC. They show that RANSAC is not guaranteed to return the optimal result. We performed extensive experiments with different data amounts (from $N=20$ to 1,000 normals), different numbers of Atlanta directions (from 2 to 6), and outlier ratios (from $p=0\%$ to 80%). Our approach always returns the globally optimal number of inliers, and this number is *always* higher than or equal to the RANSAC results.

We also show the evolution of the remaining BnB search space volume (Figs. 3(c) and 3(g)) and the number of cubes (Figs. 3(d) and 3(h)). It shows the volume drops continuously and the number of cubes remains limited. Note that the number of cubes does not have to be strictly decreasing but must remain reasonable, which is our case.

We also measure the execution time of our method with respect to the number of surface normals (Fig. 4(a)) and the number of Atlanta directions (Fig. 4(b)), and also compare it to that of the relaxed problem on the same data. With various data amounts (from $N=100$ to 100,000 normals and $M=2$), our original method shows sub-linear complexity and the relaxed one reveals efficient computation time: less than 10s even for 100,000 normals. Note that although the solution of the relaxed one for surface normals would differ

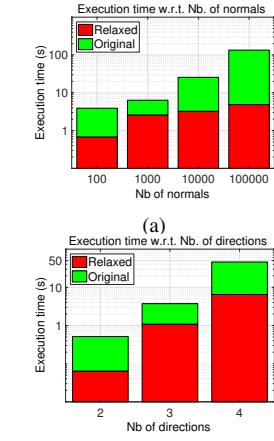


Figure 4: Execution time (log scale) of the original and relaxed problems w.r.t. (a) the number of surface normals and (b) the number of Atlanta directions.

from the solution of the original one, the number of inliers by the relaxed one is always better than or equal to the one obtained by RANSAC in all our experiments.

6.2. Real-world Dataset

We now show experimental results of our method on real-world datasets composed of lines and surface normals.

6.2.1 Line dataset

York urban database We tested our method on the York urban database [8]. It is composed of 102 images acquired in indoor and outdoor man-made environments, and provides the camera intrinsic parameters, a set of line segments manually extracted as well as the ground truth line clustering. For robustness, we removed the small line segments shorter than 5% of the image height. In addition, we included 50 outlier lines in each image. Our algorithm takes the set of lines as input and returns the AF and its associated line clustering. We applied our method to each image of the database and compared our results to the ground truth. This comparison show that, for the entire database, each line clustered by our method corresponds to its ground truth clustering, which successfully demonstrates the validity of our approach. An example is shown in Fig. 1 and additional results are available in the supplementary material.

Ricoh Theta VR database We created a novel omnidirectional VR (monoscopic) image dataset acquired by a hand-held omnidirectional Ricoh Theta camera (Fig. 5). We captured a total of 100 omnidirectional VR images with different characteristics: indoor/outdoor urban scenes, low/high number of lines, with/without vertical lines and horizontal lines, straight/tilted camera, etc. Moreover, the camera is not exactly central (no single center of projection). We extracted the lines using the method of Bazin et al. [2] on the

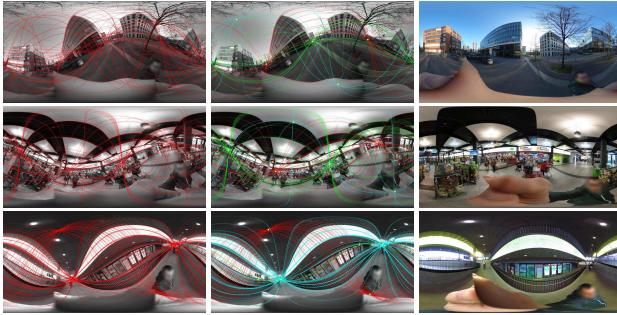


Figure 5: AF estimation results on the Ricoh Theta VR dataset. *Left*: Input lines on the images. *Middle*: Line clustering and VPs obtained by the proposed method. *Right*: Upright adjustment of the images using the estimated AF.



Figure 6: AF estimation for a different number of horizontal directions on NYUv2 dataset [28]. *Top*: A set of input scene images. *Middle&bottom*: AW segmentation based on the estimated AF for 2 and 4 horizontal directions, respectively. Each color indicates each Atlanta direction (green indicates the vertical directions and the other colors indicate the other horizontal directions).

equivalent sphere space. Our AF estimation method successfully estimates the AF as shown in Fig. 5. In addition, our method provides the vertical direction (vertical VP), and therefore it is able to rectify the vertical orientation of the pictures.² To rectify the picture, we rotate the (spherical) image such that the estimated vertical direction is mapped to the gravity direction (last column of Fig. 5). The results are visually satisfying: the images look as if they were acquired by a straight camera, the world vertical lines appear as straight vertical lines and are aligned with the y -axis of the images. This demonstrates the robustness of our method to real-world images. It also shows that our method can be successfully applied for automatic upright adjustment of omni-VR images, even with severe mis-orientations.

6.2.2 Surface normal dataset

NYUv2 dataset We qualitatively evaluated the proposed method on the NYUv2 dataset [28], which contains

²Note that the vertical direction is seamlessly estimated by our method no matter if vertical lines exist in the scene or not. In the presence of vertical lines, the vertical direction is defined by the vertical lines. In the absence of vertical lines, the vertical direction corresponds to the normal of the horizon defined by the horizontal VPs.



Figure 7: AF estimation on the large-scale 3D point cloud Bremen dataset. *Left*: A global view of the dataset. *Middle*: The estimated 6 Atlanta directions (on top-left sphere) and top-down view overlapped with Google map. *Right*: Enlarged 3D views of the top-down views (left column is input point clouds and right column is clustered point clouds). Each colored rectangle indicates a region of the top-down view.

1449 RGBD images of various indoor scenes. To validate our approach for a different number of horizontal directions (both $M=2$ and $M=4$ horizontal directions), we tested our method on the images which are labeled as multiple MF [31]. As shown in Fig. 6, our method successfully estimated AF for the different numbers of horizontal directions. Specifically, for $M=2$ horizontal directions, the estimated AF does not represent the whole man-made structure of the scene, while AF with $M=4$ represents all the supporting structures of the scene.

Bremen dataset To validate our method on a large number of normals and several Atlanta directions, we used the point clouds captured in the city center of Bremen, Germany, from Robotic 3D scan repository.³ This dataset consists of around 81 million 3D points (thus surface normals) and we considered 6 Atlanta directions in total ($M=5$). The results are available in Fig. 7. Our method successfully estimated the dominant Atlanta directions and clustered the point cloud accordingly. It shows our method can be applied on large-scale dataset composed of millions of normals and with several Atlanta directions.

7. Conclusion

We have presented a novel approach for Atlanta frame estimation. We formulate this task as an inlier set maximization problem (i.e. maximizing the number of clustered inlier normals) over the definition space of Atlanta frame. In contrast to existing techniques, our algorithm solves this inlier set maximization in a globally optimal way and is designed for Atlanta worlds. Moreover, the proposed approach can handle a different number of Atlanta directions (i.e. 2, 3 and more), be applied for a large range of images (e.g. perspective and omnidirectional) and efficiently deals with outliers. Our algorithm has been validated successfully on both synthetic data and challenging real-world dataset. Since our approach is globally optimal, we believe that our approach can be especially useful for cases where the accuracy of the returned results must be guaranteed.

³<http://kos.informatik.uni-osnabrueck.de/3Dscans/>

References

- [1] M. Antunes and J. P. Barreto. A global approach for the detection of vanishing points and mutually orthogonal vanishing directions. In *CVPR*, 2013. [1](#), [2](#)
- [2] J.-C. Bazin, C. Demonceaux, P. Vasseur, and I. S. Kweon. Motion estimation by decoupling rotation and translation in catadioptric vision. *CVIU*, 2009. [7](#)
- [3] J.-C. Bazin, C. Demonceaux, P. Vasseur, and I. S. Kweon. Rotation estimation and vanishing point extraction by omnidirectional vision in urban environment. *IJRR*, 2012. [1](#), [2](#)
- [4] J.-C. Bazin, H. Li, I. S. Kweon, C. Demonceaux, P. Vasseur, and K. Ikeuchi. A branch and bound approach to correspondence and grouping problems. *TPAMI*, 2013. [3](#)
- [5] J.-C. Bazin, Y. Seo, C. Demonceaux, P. Vasseur, K. Ikeuchi, I. Kweon, and M. Pollefeys. Globally optimal line clustering and vanishing point estimation in Manhattan world. In *CVPR*, 2012. [1](#), [2](#), [3](#)
- [6] J.-C. Bazin, Y. Seo, and M. Pollefeys. Globally optimal consensus set maximization through rotation search. In *ACCV*, 2012. [2](#)
- [7] J. Coughlan and A. Yuille. The Manhattan world assumption: Regularities in scene statistics which enable bayesian inference. In *NIPS*, 2000. [1](#), [2](#)
- [8] P. Denis, J. H. Elder, and F. J. Estrada. Efficient edge-based methods for estimating Manhattan frames in urban imagery. In *ECCV*, 2008. [1](#), [2](#), [7](#)
- [9] M. A. Fischler and R. C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. In *Communications of the ACM*, 1981. [1](#), [2](#)
- [10] A. Flint, D. Murray, and I. Reid. Manhattan scene understanding using monocular, stereo, and 3D features. In *ICCV*, 2011. [2](#)
- [11] B. Ghanem, A. Thabet, J. Carlos Niebles, and F. Caba Heilbron. Robust Manhattan frame estimation from a single RGB-D image. In *CVPR*, 2015. [2](#)
- [12] S. Gupta, P. Arbelaez, and J. Malik. Perceptual organization and recognition of indoor scenes from RGB-D images. In *CVPR*, 2013. [2](#)
- [13] R. Hartley and F. Kahl. Global optimization through rotation space search. *IJCV*, 2009. [3](#), [4](#), [5](#)
- [14] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, second edition, 2004. [1](#), [7](#)
- [15] R. Horst and H. Tuy. *Global optimization: deterministic approaches*. Springer Verlag, 2006. [4](#)
- [16] K. Joo, N. Kim, T.-H. Oh, and I. S. Kweon. Line meets as-projective-as-possible image stitching with moving dlt. In *ICIP*, 2015. [2](#)
- [17] K. Joo, T.-H. Oh, H. Kim, and I. S. Kweon. Hierarchical 3d line restoration based on angular proximity in structured environments. In *ICIP*, 2013. [2](#)
- [18] K. Joo, T.-H. Oh, J. Kim, and I. S. Kweon. Globally optimal Manhattan frame estimation in real-time. In *CVPR*, 2016. [1](#), [2](#), [6](#)
- [19] K. Joo, T.-H. Oh, J. Kim, and I. S. Kweon. Robust and globally optimal manhattan frame estimation in near real time. *TPAMI*, 2018. [2](#)
- [20] J. Kosecka and W. Zhang. Video compass. In *ECCV*, 2002. [2](#)
- [21] T. Kroeger, D. Dai, and L. J. V. Gool. Joint vanishing point extraction and tracking. In *CVPR*, 2015. [2](#)
- [22] H. Lee, E. Shechtman, J. Wang, and S. Lee. Automatic upright adjustment of photographs with robust camera calibration. *TPAMI*, 2014. [2](#)
- [23] J. Lezama, R. Grompone von Gioi, G. Randall, and J.-M. Morel. Finding vanishing points via point alignments in image primal and dual domains. In *CVPR*, 2014. [2](#)
- [24] H. Li. Consensus set maximization with guaranteed global optimality for robust geometry estimation. In *ICCV*, 2009. [3](#)
- [25] F. M. Mirzaei and S. I. Roumeliotis. Optimal estimation of vanishing points in a Manhattan world. In *ICCV*, 2011. [2](#)
- [26] G. Schindler and F. Dellaert. Atlanta world: An expectation maximization framework for simultaneous low-level edge grouping and camera calibration in complex man-made environments. In *CVPR*, 2004. [1](#), [2](#)
- [27] J. A. Shufelt. Performance evaluation and analysis of vanishing point detection techniques. *TPAMI*, 1999. [2](#)
- [28] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus. Indoor segmentation and support inference from RGBD images. In *ECCV*, 2012. [1](#), [2](#), [8](#)
- [29] J. Straub, N. Bhandari, J. J. Leonard, and J. W. Fisher. Real-time Manhattan world rotation estimation in 3D. In *IROS*, 2015. [2](#)
- [30] J. Straub, O. Freifeld, G. Rosman, J. J. Leonard, and J. W. Fisher III. The Manhattan frame model—Manhattan world inference in the space of surface normals. *TPAMI*, 2017. [2](#)
- [31] J. Straub, G. Rosman, O. Freifeld, J. J. Leonard, and J. W. Fisher, III. A mixture of Manhattan frames: Beyond the Manhattan world. In *CVPR*, 2014. [1](#), [2](#), [8](#)
- [32] C. Taylor and A. Cowley. Parsing indoor scenes using RGB-D imagery. In *Proceedings of Robotics: Science and Systems*, 2012. [2](#)
- [33] E. Tretiak, O. Barinova, P. Kohli, and V. S. Lempitsky. Geometric image parsing in man-made environments. *IJCV*, 2012. [2](#)
- [34] H. Wildenauer and A. Hanbury. Robust camera self-calibration from monocular images of Manhattan worlds. In *CVPR*, 2012. [2](#)
- [35] Y. Xu, S. Oh, and A. Hoogs. A minimum error vanishing point detection approach for uncalibrated monocular images of man-made environments. In *CVPR*, 2013. [2](#)
- [36] C. Zhang, L. Wang, and R. Yang. Semantic segmentation of urban scenes using dense depth maps. In *ECCV*, 2010. [2](#)