Computers & Graphics 38 (2014) 222–229

Contents lists available at ScienceDirect

Computers & Graphics

journal homepage: www.elsevier.com/locate/cag

Special Section on CAD/Graphics 2013

Geometric multi-covering

Rouven Strauss [a,*], Florin Isvoranu [b], Gershon Elber [a]

[a] Department of Computer Science, Technion - Israel Institute of Technology, Technion City, 32000 Haifa, Israel
[b] Evolute GmbH, Austria

ARTICLE INFO

Article history:
Received 5 August 2013
Received in revised form
3 October 2013
Accepted 19 October 2013
Available online 7 November 2013

Keywords:
Geometric multi-covering
Placement
k-Coverage
Visibility
Lighting design
Surveillance

ABSTRACT

We present a general, unified framework to resolve geometric covering problems. The problem is reduced to a set cover search in parametric space. We propose and implement different methods for solving the set cover problem, allowing for flexible trade-offs between solution quality and computation time. Our framework relies on computer graphics techniques and heavily exploits GPU based computations.

Results are demonstrated in two specific applications: firstly, multi-visibility/accessibility analysis of 3D scenes that guarantees coverage, possibly redundant, of the target shape(s) by a minimal number of observers. Secondly, illumination design in 3D environments that ensures the satisfaction of local constraints on illuminance levels using a minimal set of lamps.

© 2013 Elsevier Ltd. All rights reserved.

1. Introduction

Geometric covering (GC) problems arise in many different fields. As part of the manufacturing process, products have to be entirely validated by a minimal number of inspection devices. In scenarios involving visual surveillance, e.g., in banks or in museums, critical areas have to be visible to an as-small-as-possible set of cameras or guards, possibly more than once for redundancy (see Fig. 1). Public spaces require sufficient illumination while keeping the number of light sources minimal for energy conservation. In mold design and given a 3D artifact, a division of the artifact into a minimal set of assemblable mold parts is desired. When considering antenna networks, the objective is to achieve certain levels of service quality at different geographical locations, making it necessary to place a minimal set of antennas guaranteeing the service quality.

The aforementioned applications pose similar covering questions of geometric nature, where covering means the satisfaction of constraints on values assigned to objects in space, such as illuminance values, visibility requirements or quality-of-service levels. Due to their importance, GC questions have attracted a considerable amount of attention from various scientific disciplines such as computer graphics [1,2], computational geometry [3,4], manufacturing and mold design [5,6], surveillance [7], inspection [8] and sensing theory [9].

In this work, we introduce a framework to tackle GC problems in a general, unified way. Our approach draws heavily from computer graphics techniques and offers solutions for the above-mentioned applications as well as other GC queries. Being considered in various fields, studies about GC problems typically use different terminologies. For consistency and convenience, we first introduce the terminology used throughout this work.

Terminology: The space in which the GC problems are examined in our framework contains three types of objects:

- *targets* • *sensors* • *occluders*

An object in space that requires covering will be denoted as a *target*. Formally, a target, $\mathcal{T}$, is an m-manifold object in space, where $m \in \{1,2\}$. We pose only one constraint on $\mathcal{T}$: it must possess a parametrization from an m-dimensional box, $D_{\mathcal{T}} : [d^i_{min}, d^i_{max}]^{m-1}_{i=0}$, such that it can be represented as $\mathcal{T} = \mathcal{T}(d^0, d^1, \ldots, d^{m-1})$. For example, in $\mathbb{R}^3$, the target can be a bivariate (trimmed) surface or a set of such surfaces, but also (a set of) 3-space curves. In Fig. 1, the surface of an art pavilion serves as $\mathcal{T}$.

Any object in space that exhibits covering abilities is called a *sensor*. The sensors of a set $\mathcal{S}$ *cover* target locations, and can represent cameras or guards but also cellular antennas or light sources. Sensors are typically fixed points in space. The space in which the sensors reside is not required to equal the space in which the target resides. For example, while the target can be located $\mathbb{R}^2$, the sensors might reside in $\mathbb{R}^3$. A possible set of sensors is illustrated in Fig. 1, where 100 candidate cameras form $\mathcal{S}$.

* Corresponding author.
E-mail address: strauss@cs.technion.ac.il (R. Strauss).

0097-8493/$ - see front matter © 2013 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.cag.2013.10.018

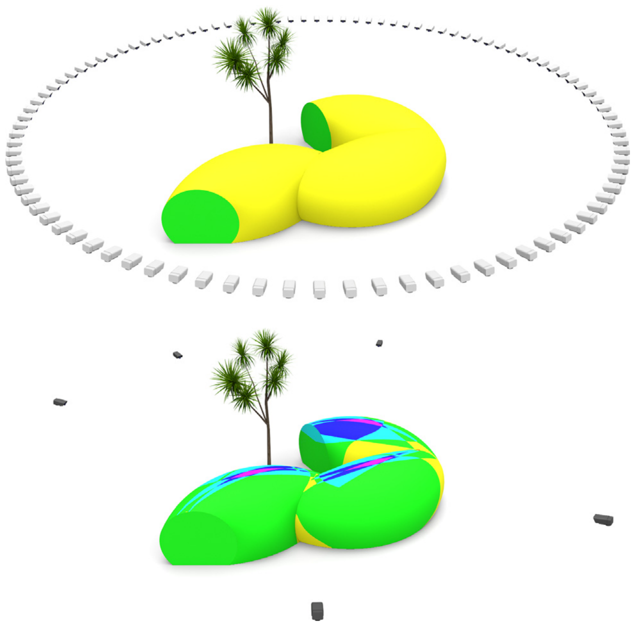

Fig. 1. Top: A surveillance scenario in which a pavilion has to be inspected. The two entrances (in green) must be visible to at least two cameras, while the hull of the pavilion (in yellow) must be covered once. One hundred candidate cameras are shown. Bottom: Five selected cameras fulfill 98.5% of the coverage constraints specified on the pavilion's surface. The different colors show the actual coverage levels (0 — red; 1 — yellow; 2 — green; 3 — turquoise; 4 — blue; 5 — magenta). (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

The third class of objects, O, are called *occluders*. Based on their location in space, occluders reduce or even prevent the coverage of target locations by sensors. For instance, in GC problems dealing with visibility, occluders may be opaque objects blocking any inspection through them. In Fig. 1, a tree serves as an occluder.

The sensors cover target locations with a certain *coverage level*. In turn, every target location has a *required coverage level* assigned to it. Based on the provided terminology, GC problems thus reduce to the question of *what is the best placement of a* (*minimal*) *number of sensors such that the required coverage levels are satisfied for all target locations*.

Certain GC problems share the property that the coverage levels are binary. For instance, in applications concerned with visibility, the coverage levels encode the states 'visible' and 'invisible'. We refer to such GC problems as *Binary Geometric Covering* (BGC) problems. However, in many GC problems, the covering is not necessarily binary. For instance, certain target locations may be required to be visible by several cameras, offering some redundancy. Similarly, multiple light sources may accumulatively illuminate target points to reach the desired illumination levels. A GC problem where the required coverage levels are above $\mathbb{R}$ is, therefore, denoted as a *Continuous Geometric Covering* (CGC) problem. Analogously, any GC problem in which the required coverage levels are above the natural numbers is called a *Discrete Geometric Covering* (DGC) problem.

Our approach: The vast majority of previous work on GC problems explore the solution in the space in which the sensors and targets are situated, typically in 2D or 3D Euclidean space. In contrast, we propose an approach that reduces GC problems into a generic problem in the parametric domain of the target, $D_{\mathcal{T}}$. The problem is then discretely solved, exploiting its highly parallel nature and using computer graphics techniques. Despite its discrete and therefore approximate character, our approach offers a simple, robust, and unified framework to address a large variety of GC problems, including the aforementioned ones. Furthermore, by reducing the problem to many simple (i.e., pixel) problems, we are able to handle and support *local* covering specifications. That is, any location on the target can have its own required coverage level. To the best of our knowledge, existing GC algorithms can only handle the global specification of a coverage requirement, i.e., achieving the same covering level for all target locations.

GC problems are considered difficult to solve. As stated, the majority of known GC algorithms operate in Euclidean space and, as part of the solution process, must resolve complex visibility and accessibility queries among different 3D objects, typically polygons. Moreover, GC problems are typically reduced to a set cover query and hence of expected exponential time complexity. In this work, we propose different methods for answering the set cover queries, each involving a different trade-off regarding solution quality, runtime and extensibility. For higher efficiency, we take advantage of GPU computation capabilities wherever possible.

Organization: The rest of this work is organized as follows: after discussing previous work concerned with different types of GC problems in Section 2, we provide a formal definition of GC problems in Section 3. In Section 4, we present our algorithmic approach for computationally solving GC problems. Two different computer graphics related GC applications are discussed in Section 5: multi-visibility analysis by cameras or guards and illuminance satisfaction by multiple light sources. Finally, we discuss the proposed approach and possible extensions in Section 6, and conclude this work in Section 7.

2. Related work

One of the classical GC problems is arguably the famous Art Gallery Problem [7]. While an extensive amount of GC problem variations has since been addressed in literature, we restrict our discussion mostly to related work solving problems in three dimensions. Following [10], we refer to problems requiring a globally defined covering level k as k-coverage problems.

Inspection: Tarbox and Gottschlich [11] sample sensors on a sphere surrounding a three-dimensional target and choose a set of locations on the surface of the target. Different heuristic-based algorithms are used to find a small, but not necessarily minimal set of sensors such that all target locations are 1-covered. Restricting themselves to simple polyhedra, Roberts and Marshall [8] attempt to find sets of faces of a given target in $\mathbb{R}^3$ that are 1-covered by a common viewpoint while minimizing the number of viewpoints.

Computer graphics: Fleishman et al. [1] tackle a BGC problem aiming at automatically creating a small number of rendered images of a given 3D scene such that most surfaces in the scene are 1-covered. They furthermore require that any surface be covered at most once, in order to obtain non-redundant images.

Mold design: In mold design, 1-cover must be precisely satisfied, i.e., every target location belongs to exactly one mold part. Liu and Ramani [5] process the geometry in Euclidean space, while Shragai and Elber [6] work in parametric space. The latter work is similar to our approach and can be seen as a special case of it.

Sensor networks: Lu et al. [9] assign to every sensor s a probability function mapping every location in $\mathbb{R}^2$ to a probability with which it is detected by s. All required coverage levels are the same, representing the expected number of sensors detecting the corresponding target location. The resulting k-coverage problem is solved using a greedy method. Becker et al. [12] consider 1-coverage of every voxel of a 3D volume, employing a greedy algorithm to select a small number of sensors.

Lighting design: GC problems also arise in lighting design that seeks to automatically place lights such that a prescribed illumination is achieved. The problem of approximating a predefined illumination setting is tackled by Schoeneman et al. [2], Marschner

Download English Version:

https://daneshyari.com/en/article/441943

Download Persian Version:

https://daneshyari.com/article/441943

Daneshyari.com