Journal of Cultural Heritage xxx (2015) xxx–xxx

Available online at
ScienceDirect
www.sciencedirect.com

Elsevier Masson France
EM|consulte
www.em-consulte.com/en

Original article

Modeling a virtual robotic system for automated 3D digitization of cultural heritage artifacts

Antonio De Stefano [a,*], Reimar Tausch [b], Pedro Santos [b], Arjan Kuijper [b,c], Giuseppe Di Gironimo [a], Dieter W. Fellner [b,c], Bruno Siciliano [d]

[a] Department of Industrial Engineering (DII), University of Naples Federico II, P. le Tecchio 80, Naples, Italy
[b] Competence Center for Cultural Heritage Digitization (CC CHD), Fraunhofer IGD, Fraunhoferstr. 5, Darmstadt, Germany
[c] Interactive Graphics Systems Group (GRIS), Technische Universität Darmstadt, Fraunhoferstr. 5, Darmstadt, Germany
[d] Department of Electrical Engineering and Information Technology (DIETI), University of Naples Federico II, Via Claudio 21, Naples, Italy

ARTICLE INFO

Article history:
Received 21 April 2015
Accepted 13 November 2015
Available online xxx

Keywords:
Robotic arm
Automation
Inverse kinematics
View planning
3D scanning

ABSTRACT

Complete and detailed 3D scanning of cultural heritage artifacts is a still time-consuming process that requires skilled operators. Automating the digitization process is necessary to deal with the growing amount of artifacts available. It poses a challenging task because of the uniqueness and variety in size, shape and texture of these artifacts. Scanning devices have usually a limited focus or measurement volume and thus require precise positioning. We propose a robotic system for automated photogrammetric 3D reconstruction. It consists of a lightweight robotic arm with a mounted camera and a turntable for the artifact. In a virtual 3D environment, all relevant parts of the system are modeled and monitored. Here, camera views in position and orientation can be planned with respect to the depth of field of the camera, the size of the object and preferred coverage density. Given a desired view, solving inverse kinematics allows for collision-free and stable optimization of joint configurations and turntable rotation. We adopt the closed-loop inverse kinematics (CLIK) algorithm to solve the inverse kinematics on the basis of a particular definition of the orientation error. The design and parameters of the solver are described involving the option to shift the weighting between different parts of the objective function, such as precision or mechanical stability. We then use these kinematic solutions to perform the actual scanning of real objects. We conduct several tests with different kinds of objects showing reliable and sufficient results in positioning and safety. We present a visual comparison involving the real robotic system with its virtual environment demonstrating how view poses for different-sized objects are successfully planned, achieved and used for 3D reconstruction.

1. Introduction

The recent progress in 3D digitization of cultural heritage artifacts is motivated by various reasons. Among these, risk of complete and irrecoverable loss of cultural heritage due to destruction or decay, e.g. caused by catastrophes and wars, is perhaps the most critical one. Furthermore, the digital copy can support the preservation and restoration of the original objects by minimizing the physical interaction and enabling virtual planning.

However, even for smaller artifacts the process of complete and detailed 3D digitization can be tedious and time-consuming while requiring specific knowledge and skills. In view of the growing amount of objects eligible for scanning, automatizing the process becomes essential. The *CultLab3D* [1] digitization pipeline (Fig. 1) aims at automated and economical mass digitization in 3D comprising geometry, texture and even optical material properties. By deploying novel robotic and scanning technologies, human interaction with artifact and scanning devices is reduced to a minimum.

A similar approach can be found in automated industrial quality inspection using optical scanners. Those systems are usually specialized on a certain series of identical products and benefit from available CAD models for the scan planning and comparison [2].

* Corresponding author at: Department of Industrial Engineering (DII), University of Naples Federico II, P. le Tecchio 80, Naples, Italy.
E-mail addresses: antonio.destefano6@studenti.unina.it, a.destefano86@gmail.com (A. De Stefano), reimar.tausch@igd.fraunhofer.de (R. Tausch), pedro.santos@igd.fraunhofer.de (P. Santos), arjan.kuijper@igd.fraunhofer.de (A. Kuijper), giuseppe.digironimo@unina.it (G. Di Gironimo), dieter.fellner@gris.tu-darmstadt.de (D.W. Fellner), bruno.siciliano@unina.it (B. Siciliano).

http://dx.doi.org/10.1016/j.culher.2015.11.008
1296-2074/

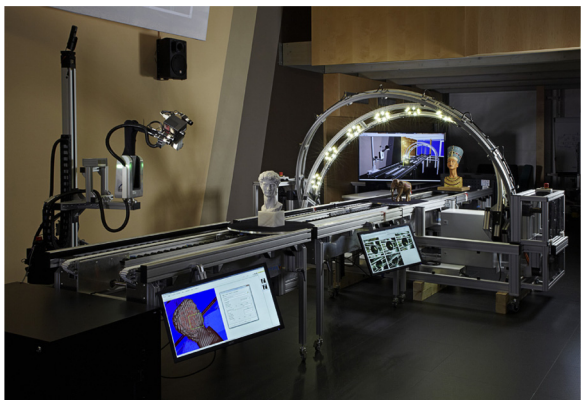

Fig. 1. An overview of the *CultLab3D* digitization pipeline [1]. Cultural heritage artifacts up to a maximum diameter of 60 cm are placed on conveyor belts and scanned at turntables with a 3D scanning camera and light arcs as well as with lightweight 3D scanning robot arms.

On the contrary, in the cultural heritage domain the unique objects come in a large variety regarding size, shape complexity and texture. This calls for a flexible scanning setup which is able to adapt itself to the objects of interest.

The *CultArm3D* module of *CultLab3D* system is such a flexible scanning setup, consisting of a flexible lightweight robotic arm with a mounted camera or scanner and a turntable for the object. The module is designed to position the scanning device and achieve various viewing angles to the object in order to complete surface coverage and resolve possible occlusions.

This paper addresses the problem of planning and achieving scan views with respect to the size of the object of interest, where the focus is on how the desired views translate to the robotic system. We therefore introduce an inverse kinematics solver that estimates the joint configurations and rotation angle. Thus, different viewing candidates can be evaluated and compared regarding their reachability, stability, safety and precision. After the inverse kinematics solver has found a solution, multiple theoretical solutions for the same view pose can be found since the structure of the manipulator is known. They create a set of available configurations which keep the camera view unaltered. The set of multiple configurations for a given scan view is then used to improve collision avoidance and stability of the manipulator. Several tests with different kinds of objects show reliable and sufficient results in positioning and safety. A visual comparison of the real robotic system with its virtual environment shows that view poses for different-sized objects are indeed successfully planned and achieved.

This paper is organized as follows. In Section 2, we describe the setup and environment of the robotic system followed by related work in Section 3. We then explain our approach in Section 4 and show experiments and results in Section 5. Finally, we describe conclusions and future work in Section 6.

2. Setup and environment

The *CultLab3D* digitization pipeline [1] is composed of hardware and software components; among these are conveyor belts and turntable, 3D scanning camera and light arcs as well as lightweight 3D scanning robot arms. The goal is to provide a 3D mass digitization solution for most cultural heritage artifacts up to a maximum diameter of 60 cm.

User interaction is limited to setting up target objects on carrier tablets, and picking them up again after acquisition. During the digitization process the artifacts move along a fully automated conveyor system and pass several scanning stations specialized for capturing certain visual object properties.

The basic photogrammetric configuration of the digitization pipeline consists of two scanning stations. At the first scanning station, a self-sufficient mechanism called *CultArc3D* captures geometry, texture and optical material properties using a motorized camera and light arc, featuring industrial, high-resolution video cameras. While *CultArc3D*'s advantage is the fast and parallel acquisition, it cannot resolve self-occlusions on complex-shaped objects, since its cameras and light sources are fixed at certain mounting points on rotary half-circular arcs. Thus, their motion and viewing positions are limited to a hemisphere with the object of interest in the center.

The aforementioned issue is addressed by the second scanning station called *CultArm3D* which is the scope of this article.

2.1. CultArm3D

Using the information of the 3D model draft obtained from the first scanning station, an iterative scan plan can be calculated to find a set of scan views to completely resolve the remaining uncertainties and self-occlusions in the final 3D model (Fig. 2)

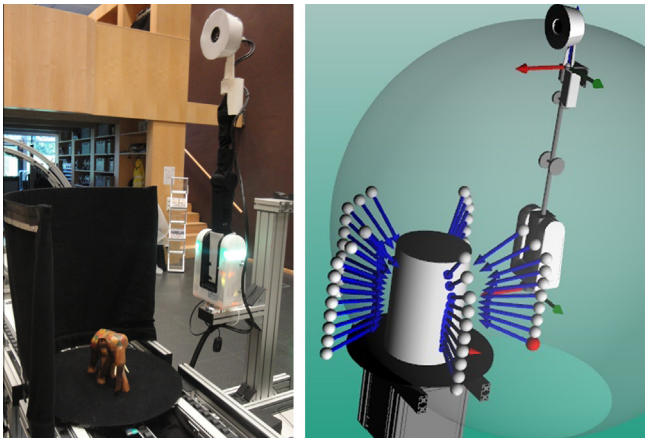

Fig. 2. Real and virtual setup of the *CultArm3D* module.

The *CultArm3D* module shown on Fig. 2 is designed to achieve these missing views in position and orientation. It combines a bionically inspired, lightweight robotic arm [3] focused on safety with a turntable for the object. The compliant robotic arm consists of 5 degrees of freedom (DOFs) with all rotational joints and can accurately move payloads up to 2 kg. The turntable is capable to precisely and smoothly rotate, clockwise and counterclockwise, around its vertical axis with infinite 360 degree rotations and is designed for object weights up to 50 kg. A high-resolution (10–20 mega pixels) camera together with a diffused white light source (LED ring) is mounted at the end of the robotic arm. The ability of the robotic system to achieve close-up views to the object, as well as the higher focal length of the camera lens allows for more detailed photography compared to the previous scanning station *CultArc3D*. Thus, even smaller and previously inaccessible features can contribute to the 3D reconstruction and locally enhance the model quality in coverage, geometry and texture resolution. However, complicated surfaces involving blank, shiny or translucent materials are problematic to captured with the current lighting setup and the standard photogrammetric 3D reconstruction process.

Please cite this article in press as: A. De Stefano, et al., Modeling a virtual robotic system for automated 3D digitization of cultural heritage artifacts, Journal of Cultural Heritage (2015), http://dx.doi.org/10.1016/j.culher.2015.11.008

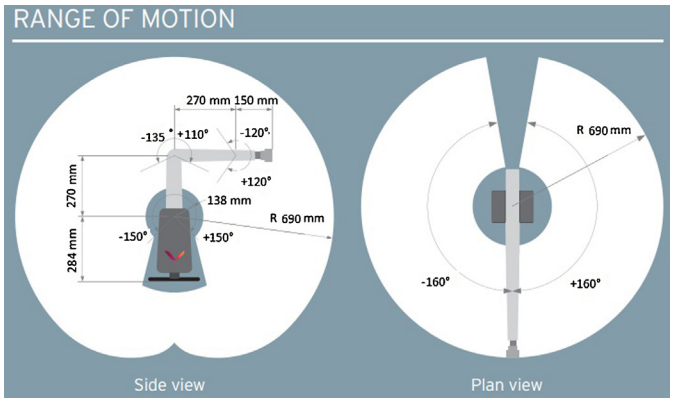

Fig. 3. Side and plan view of the work space [3].

The quasi-hemispheric work space of the robotic arm shown on Fig. 3 suggests a mounting point for the robot base aside of a turntable in order to increase the number of reachable scan views. This poses the challenging task of synchronizing and controlling each part of the *CultArm3D* setup in a way that desired scan views are achieved with sufficient precision, i.e. not every scan view that would be optimal in terms of model coverage is exactly feasible by the kinematic system. Multiple safety constraints have to hold while trade-offs in precision of position, distance and orientation of the scan view can be modeled. In the following applicable modeling and solver strategies are presented and discussed.

3. Related work

Automated 3D reconstruction using robots with mounted scanners has been addressed in several recent studies [4–6]. However, while applying heavy-duty industrial robots with oversized workspaces, the focus is mostly on surface geometry-based planning of views, rather than having to care about the kinematics of an abstract 6D positioning system. The problem of kinematic accessibility becomes prevalent when applied to a lightweight collaborative robot with a very limited workspace to the task, as in our approach. Therefore, on solutions to the inverse kinematics (IK) problem and give a brief overview of present IK solver techniques related to the task.

The field can be divided into two classes of techniques, the geometric/analytical exact methods [7] and iterative methods, such as cyclic coordinate descent methods [8,9], differential-based methods [10], combined optimization methods [11], and neural network methods [12], to mention only some.

The geometric/analytical algorithms perform closed-form solutions and tend to be fast because they reduce the IK problem to a set of mathematical equations that need only to be evaluated in a single step to produce a result but turn out to suffer from poor scalability. The limitations of this class of solvers are strictly related to the specific robot configuration which can yield to different approaches even if the task is similar [13].

IK solvers based on cyclic coordinate descent methods (CCD) use an iterative approach which takes multiple steps towards a solution. The steps that the solver takes are heuristically formed. Because the iterative step is heuristically driven, accuracy is normally the price paid for speed.

Like the CCD technique, differential-based techniques utilize an iterative approach that requires multiple steps to find a solution. The steps made by the algorithm are determined considering the Jacobian matrix that virtually applies small changes in joint configuration to positional offsets. Since all the joint angles are updated in a single step, the movements are dissipated over the whole chain which results in a more realistic looking posture. This class includes methods, such as resolved motion rate control [14] and closed-loop inverse kinematics [10,15].

Optimization methods like the conjugate gradient method or quasi-Newton method can be used as approximation when the Jacobian matrix, necessary for the previous methods, is too expensive to compute at each iteration or is ill-conditioned.

Artificial neural networks have been used to solve a wide variety of tasks that are hard to solve using ordinary rule-based programming. This case can also be applied to robot manipulators [16].

Some traditional inverse kinematic solvers for 5-DOF manipulators [17] are not applicable in this case due to their fundamental condition for reachability of the target pose given by the user. In this case however, scan views created by a solely object surface-based view planning are not exactly reachable (in position and orientation) by the robotic setup, nor can be easily checked for reachability as stated in [18]. For this reason, an iterative approach has been chosen in which an error representation can be modeled in order to always retrieve the closest possible solution for a target pose given as input.

A differential method has been chosen because once the geometrical structure of the robot is known, the geometrical Jacobian matrix, which is the core of the differential approach, can be easily calculate with a numerical procedure[19]. With the CCD method an heuristic solution would be found not providing an optimal solution. It can be demonstrated, using the Lyapunov method [20], that the so-called closed-loop inverse kinematics (CLIK) algorithm [15,19] assure asymptotic stability of the error system and thus an optimal solution. As counterpart, the error dynamics is governed by a nonlinear differential equation. An artificial neural network require machine learning which can be dangerous and time-consuming when dealing with a wide variety of objects. Furthermore, the implementation of the code would require a bigger effort and an higher computational time.

For these reasons, the CLIK algorithm has been chosen which attempts to minimize the generalized error (in position and orientation). By regulating the thresholds for positions and orientation the accuracy of the solution with respect to the desired one can be estimated.

4. Approach and implementation

As seen in Section 3, a variety of implementations are available to solve the inverse kinematic problem. Each approach has its pros and cons depending on the structure of the robotic arm and on the required task. For this application, a differential-based iterative method has been preferred in order to implement a robust and efficient solution using a CLIK algorithm. This technique is well-established and computationally effective when the Jacobian transpose method is used [10] as in this case.

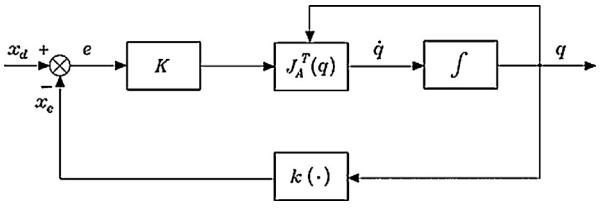

Fig. 4. Scheme for closed-loop inverse kinematic with the Jacobian transpose [19].

The CLIK scheme with the usage of the Jacobian matrix is shown on Fig. 4 and visualizes the iterative operation characteristic of the system, detailed as follows:

- as first step an initial trial pose is given as solution seed $\boldsymbol{x}_e = \boldsymbol{x}_{e_{\text{seed}}}$ from which the algorithm starts. The initial pose must be non-singular and within the workspace of the robot;

Please cite this article in press as: A. De Stefano, et al., Modeling a virtual robotic system for automated 3D digitization of cultural heritage artifacts, Journal of Cultural Heritage (2015), http://dx.doi.org/10.1016/j.culher.2015.11.008

- the error $\boldsymbol{e}$ is calculated with respect to the desired pose $\boldsymbol{x}_d$ and the current approximate pose $\boldsymbol{x}_e$, see Section 4.1 for details;
- check a termination criteria regarding an error threshold, a maximum number of iterations, or a required solution difference. If the termination criteria is met, end the process with the current $\boldsymbol{x}_e$ as solution, else continue;
- the error is properly weighted through a matrix gain $\boldsymbol{K}$, see Section 4.2;
- using the error $\boldsymbol{e}$ and the Jacobian transpose matrix $\boldsymbol{J}_A^T(\boldsymbol{q})$ the algorithm estimates a new possible joint velocity configuration $\dot{\boldsymbol{q}}$ for error reduction;
- with the integration block the new joint configuration $\boldsymbol{q}$ is retrieved from the joint velocity $\dot{\boldsymbol{q}}$;
- through the direct kinematics expression $k(\cdot)$ and the joint position configuration $\boldsymbol{q}$ the approximate pose $\boldsymbol{x}_e$ is obtained, see Section 4.3. Continue with step 2.

Given a set of desired poses, e.g. obtained from a view planning algorithm, the goal of the application is to minimize the error between the desired and the actually achievable camera pose. The pose comprises position and orientation where only the view direction is important, i.e. the rotation around the view vector (camera roll) is negligible.

4.1. Definition of pose error

In order to evaluate how different the iteratively calculated pose is from the desired one, an adequate function to measure the pose error has to be defined. While it is straightforward to define the position error $\boldsymbol{e}_p$ as the difference between the actual position $\boldsymbol{p}_e(\boldsymbol{q})$ and the desired position $\boldsymbol{p}_d$ in operational space as

$$\boldsymbol{e}_p = \boldsymbol{p}_d - \boldsymbol{p}_e\{\boldsymbol{q}\} \tag{1}$$

where $\boldsymbol{q} = \{q_i\}_{i=1,\cdots,5}$ represents the vector containing all the joint variables q_i of the robot setup. More attention has to be paid to the definition of the orientation error, which can be formulated after deciding about the mathematical representation of orientation itself suitable for the given task and environment. In this work, an angle and axis representation has been suitably chosen in order to avoid explicit representation of singularities and relative matrix calculation, and furthermore keep compatibility with the results of previously designed and applied view planning systems [1].

A canonical way to define the orientation error with an angle and axis representation is given by the expression

$$\widehat{\boldsymbol{e}_{\boldsymbol{o}}} = 1/2\left\{\widehat{\boldsymbol{n}_e}\{\boldsymbol{q}\}\times\widehat{\boldsymbol{n}_d} + \widehat{\boldsymbol{s}_e}\{\boldsymbol{q}\}\times\widehat{\boldsymbol{s}_d} + \widehat{\boldsymbol{a}_e}\{\boldsymbol{q}\}\times\widehat{\boldsymbol{a}_d}\right\} \tag{2}$$

where $\widehat{\boldsymbol{n}_e}$ indicates the x-versor of the reference frame attached to the camera and $\widehat{\boldsymbol{n}_d}$ the x-versor of the reference frame representing the desired pose. Correspondingly, $\widehat{\boldsymbol{s}_e}$, $\widehat{\boldsymbol{s}_d}$ and $\widehat{\boldsymbol{a}_e}$, $\widehat{\boldsymbol{a}_d}$ indicate the y and z-versors of reference and desired poses, respectively. For the particular task, only the displacement between $\widehat{\boldsymbol{a}_d}$ and $\widehat{\boldsymbol{a}_e}\{\boldsymbol{q}\}$ is considered. Consequently, the orientation error can be expressed as follows

$$\widehat{\boldsymbol{e}_{vo}} = 1/2\left\{\widehat{\boldsymbol{a}_e}\{\boldsymbol{q}\}\times\widehat{\boldsymbol{a}_d}\right\} \tag{3}$$

Now that both, position and orientation errors are defined, the overall error can be expressed as

$$\boldsymbol{e} = \begin{bmatrix} \boldsymbol{e}_p \\ \widehat{\boldsymbol{e}_{vo}} \end{bmatrix} \tag{4}$$

4.2. Matrix gain

The (3×3) diagonal matrices $\boldsymbol{K}_p$ and $\boldsymbol{K}_e$ are introduced to weight the position error $\boldsymbol{e}_p$ and the orientation error $\widehat{\boldsymbol{e}_{vo}}$, respectively. The matrices are combined as follows

$$\boldsymbol{K} = \begin{bmatrix} \boldsymbol{K}_p & 0 \\ 0 & \boldsymbol{K}_e \end{bmatrix} \tag{5}$$

to form the gain matrix $\boldsymbol{K}$ indicated on Fig. 4 to weight the generalized error $\boldsymbol{e}$. This error is iteratively evaluated with respect to the updating rule for the inverse kinematics algorithm.

4.3. View pose generation

Once the inverse kinematics system of *CultArm3D* is modeled, a set of target poses have to be generated for testing purposes. Generally, these poses can be iteratively calculated during the scanning, based on the progressing digitized 3D model by utilizing NBV techniques. Since this is not the scope of this paper and for the sake of simplicity, a complete set of target poses is generated using only six parameters which refer to camera properties, such as focus distance and aperture angle, and prior object knowledge represented by bounding cylinder. A first example of resulting view radially pointing at a virtual cylinder can be observed on Fig. 2. The parameters are listed as follows and defined in the experiments in Section 5:

- *object diameter*: diameter of the bounding cylinder surrounding the object of interest;
- *object height*: height of the bounding cylinder surrounding the object of interest;
- *camera distance*: optimal distance from camera to bounding cylinder surface;
- *ground distance*: minimum distance from camera to turntable plane;
- *view levels*: number of scan view levels along the vertical axis;
- *views per level*: number of scan views per level.

In Section 5, this parameter set is used to configure the experiments.

4.4. Stability of solutions

In order to measure the stability $w(\boldsymbol{q})$ of a certain joint configuration $\boldsymbol{q}$, a function depending on joint values and their related mechanical limits is created. In this application a configuration is considered stable if the compliant robot arm is not too contracted nor too extended and thus its joint values are moderate and not near to their mechanical limits. With this index we are able to choose the configuration which has the less number of joints close to the limits and so easy to recover and fast to move to another pose. Therefore, the following function is introduced [19]:

$$w\{\boldsymbol{q}\} = -\frac{1}{2n}\sum_{i=1}^{n}\left\{\frac{q_i - \bar{q}_i}{q_{i_{MAX}} - q_{i_{\min}}}\right\}^2 \tag{6}$$

where $q_{i_{MAX}}$ denotes the i-th maximum joint limit, $q_{i_{\min}}$ denotes the i-th minimum joint limit and $\bar{q}_i$ denotes the most stable middle value of the i-th joint range. The higher the differences from these middle values sum up, the lower is the stability of the configuration. Maximizing this index yields the closest configurations to the middle joint positions and in turn a stable behavior.

4.5. Calculation of turntable angle

The purpose of the turntable is to rotate the object of interest together with the attached target views and bring them into the

Please cite this article in press as: A. De Stefano, et al., Modeling a virtual robotic system for automated 3D digitization of cultural heritage artifacts, Journal of Cultural Heritage (2015), http://dx.doi.org/10.1016/j.culher.2015.11.008

Table 1
Parameter sets for the view pose generation used in the experiments with small, medium and large-sized objects.

	Model size		
parameters	Small	Medium	Large
Object diameter	80 mm	300 mm	400 mm
Object height	400 mm	250 mm	600 mm
Camera distance	60 mm	100 mm	300 mm
Ground distance	70 mm	50 mm	70 mm
View levels	3	5	10
Views per level	5	10	5

unconstrained working space of the robot arm. Furthermore, the aim is not only to achieve the view pose but also to reduce the stress and increase the stability of the compliant robotic arm by seeking for stable joint configurations.

Besides physical constraints, which have to be introduced in order to avoid robot self-collision or collision with the object or turntable, only an approximately 120 sector of a circle is accessible at the turntable. This environmental constraint is introduced because of a cylindrical curtain partially enclosing the turntable and shielding camera views from unwanted background details (Fig. 2). Thus, the photogrammetric 3D reconstruction becomes feasible and can be reduced to only the object of interest.

However, the free operational space for the robot arm is reduced accordingly, which makes it difficult to find a sufficient combination of turntable and robot arm configurations. Given a target view, the following necessary steps are needed to calculate a suitable turntable rotation angle:

- project the target view vector on the surface plane of the turntable to reduce the problem;
- bring the vector to the middle accessible sector of the circle by virtually rotating the turntable;
- starting from this position subdivide the sector using a predefined angle offset and create an ordered set of turntable rotations to virtually traverse;
- try to solve the inverse kinematics of the robot arm for the virtually rotated target view vector;
- save the joint configuration and the evaluated overall error involving position, orientation and joint relaxation/stability as described in Section 4.4;
- if there are remaining turntable rotations to traverse, virtually apply the next one and continue at step 4;
- choose the best found configuration with respect to the overall error and apply it to the real setup.

Normally, this procedure has to be applied for each of the planned target views; where it is convenient to subsequently process views spatially adjacent to each other and reuse previously found configurations as start for searching the next one. This can further reduce the overall scanning time.

5. Experiments and discussion

Our approach was tested for three kinds of objects with different parameters regarding object dimensions and camera properties. Small, medium, and large-sized objects were taken in consideration. The experiments were carried out using testing objects and replica artifacts. In the following figures a visual comparison between planned and achieved poses will be given. On the left side the virtual planning environment is shown. Small spheres and arrows indicate respectively desired positions and orientations of the camera and the object is represented by the two parameters height and diameter forming a bounding cylinder. On the right side the real setup is shown with the achieved robot pose near the object.

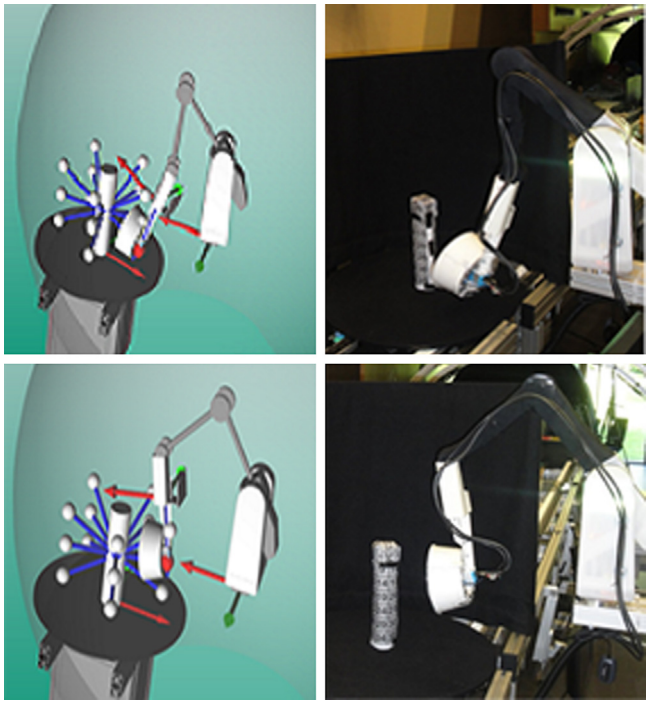

Fig. 5. Small-sized object at first and second scan level.

5.1. Example for small-sized objects

The views were generated using the parameter set in the column of Table 1 for a small and thin test object. Resulting configurations at different view levels are shown on Fig. 5. This case demonstrates how close the robot with its mounted camera and light source can get to the object. The joint configurations were considered stable and the transitions between the view levels were executed safely. The position errors for this test were in the range from 1 mm up to 5 mm.

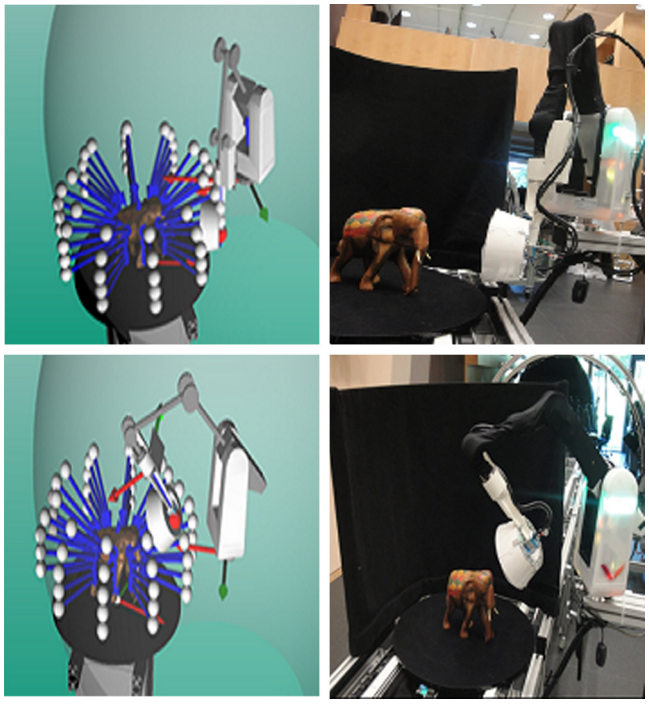

Fig. 6. A medium-sized object at first and last level.

Please cite this article in press as: A. De Stefano, et al., Modeling a virtual robotic system for automated 3D digitization of cultural heritage artifacts, Journal of Cultural Heritage (2015), http://dx.doi.org/10.1016/j.culher.2015.11.008

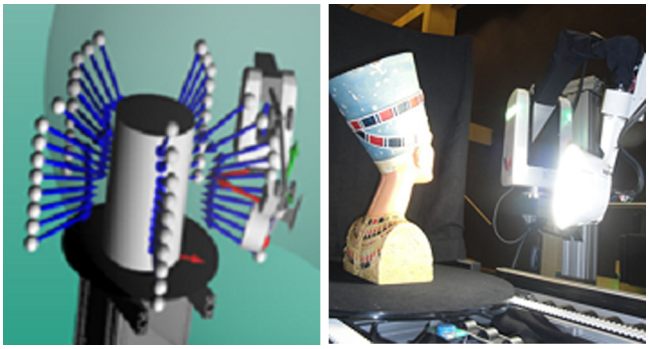

Fig. 7. A large-sized object at second scan level.

5.2. Example for medium-sized objects

Using parameters in the corresponding column of Table 1 views were generated for a more complex-shaped medium-sized object. Fig. 6 shows results of calculated joint configurations at different view levels. To ensure an accurate and detailed 3D reconstruction the scanning coverage has to be increased by generating a higher number of views. Although the difference in position and orientation between views of adjacent levels decreased, completely different corresponding joint configurations were observed in some cases. This can lead to more time-consuming transitions. A relocation of the robot base enabling more workspace for larger objects was found to reduce this problem.

5.3. Example for large-sized objects

For large-sized objects views were generated using parameters in the corresponding column of Table 1. A planned and achieved joint configuration for scanning a replica of the Nefertiti bust can be observed on Fig. 7. Some of the resulting photos that are used for the 3D reconstruction process are presented on Fig. 8. It can be observed how close views of the object change with rotation and level. The object is shown upside down because of the way the robot holds the camera, but this does not impair the photogrammetric reconstruction. More important is to shield the object from not relevant background consequently leading to an accurately segmented 3D model shown on Fig. 9. This model was reconstructed from the complete planned view set of 50 photos. The total scanning process has taken 4 min 20 s with an average of approx. 5 s per positioning plus photo. However, for objects with complex concave surface structures, such as cavities, the number of planned views should be increased to maintain the coverage and overlap, which is important for the reconstruction process. Furthermore, instead of representing the object simply as bounding cylinder, adaptive surface-based view planning techniques should then be used. That, however, exceeds the scope of this article.

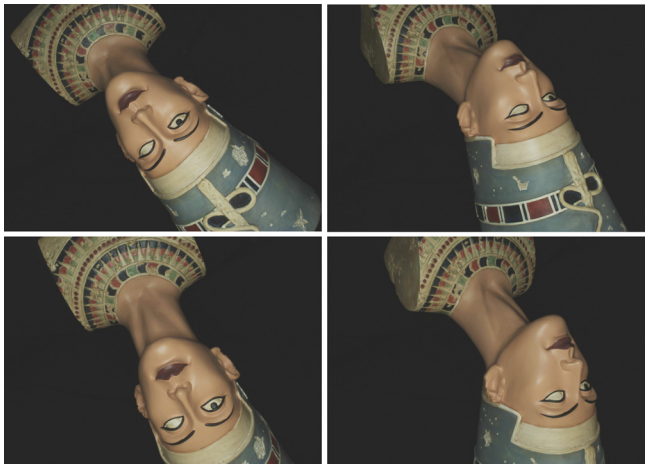

Fig. 8. Exemplary photos used for the 3D reconstruction.

Fig. 9. Mesh and textured 3D model of the Nefertiti bust.

In general, the calculation of joint configurations for large objects with diameters and heights up to 600 mm plus an optimal camera distance of 300 mm, pose a challenging task considering the limited workspace (Fig. 3) of the robot. However, an adequate placement of the robot base combined with the usage of the turntable helped to achieve all views generated for small to large-sized objects and thus ensure accurate photo quality for the reconstruction process. Concerning the computation time our implementation of the CLIK algorithm runs fast enough to either calculate robot joint configurations for all planned views in advance or successively in real-time during the scanning process.

6. Conclusions and future work

In this paper we address the open issue of automated and detailed 3D scanning of cultural heritage artifacts. As scanning devices usually have a limited focus or measurement volume and thus require precise positioning, we proposed a robotic system for photogrammetric 3D reconstruction consisting of a lightweight robotic arm with a mounted camera and a turntable for the object. We developed a virtual environment for the *CultArm3D* photogrammetric scanning module comprising a robotic arm and a turntable. We approached the problem of scan pose planning by adapting an inverse kinematics solver applied in a closed-loop inverse kinematics scheme (Fig. 4). While a variety of environmental constraints have been taken into account, we have successfully found adequate solutions in the form of combined joint configurations for robot arm and turntable. The design and parameters of the solver are described involving the option to shift the weighting between different parts of the objective function, such as precision or stability. We conducted several tests with different-sized objects demonstrating reliable results in positioning precision and accurate photo quality for 3D reconstruction.

Please cite this article in press as: A. De Stefano, et al., Modeling a virtual robotic system for automated 3D digitization of cultural heritage artifacts, Journal of Cultural Heritage (2015), http://dx.doi.org/10.1016/j.culher.2015.11.008

In the future, we plan to extend the evaluation of experiments to alternative mounting positions of the robotic arm and more complex objects yielding various non-radially pointing view vectors. While in most cases the transition trajectory between two poses could be sufficiently achieved with a simple interpolation between the joint configurations, more complex objects and view vectors will also require to explicitly plan collision-free motion trajectories. Also regarding the minimization of the overall required scanning time for an object, previously achieved views should be used as seed input for the kinematics algorithm to resolve next views. Finally, we want to stronger combine kinematic as well as 3D geometry-based view planning aspects into one optimization problem for a time-minimal and accurate scanning process.

Acknowledgements

The work of this study arises from a collaboration between the Competence Center of Cultural Heritage Digitization (CC CHD) of Fraunhofer Institute for Computer Graphics Research (IGD) and the Department of Industrial Engineering (DII) of University of Naples Federico II.

References

[1] P. Santos, M. Ritz, R. Tausch, H. Schmedt, R. Monroy, A. De Stefano, O. Posniak, C. Fuhrmann, D.W. Fellner, CultLab3D – on the verge of 3D mass digitization, in: R. Klein, P. Santos (Eds.), Eurographics Workshop on Graphics and Cultural Heritage, The Eurographics Association, 2014, pp. 65–73.
[2] S. Kahn, U. Bockholt, A. Kuijper, D.W. Fellner, Towards precise real-time 3D difference detection for industrial applications, Comput. Ind. 64 (9) (2013) 1115–1128.
[3] T. Lens, J. Kunz, O. von Stryk, C. Trommer, A. Karguth, BioRob-Arm: a quickly deployable and intrinsically safe, lightweight robot arm for service robotics applications, in: Robotics (ISR), 2010 41st International Symposium on and 2010 6th German Conference on Robotics (ROBOTIK), 2010, pp. 1–6.
[4] S. Kriegel, T. Bodenmüller, M. Suppa, G. Hirzinger, A surface-based Next-Best-View approach for automated 3D model completion of unknown objects, in: Robotics and Automation (ICRA), 2011 IEEE International Conference on, 2011, pp. 4869–4874.
[5] W.R. Scott, G. Roth, J.-F. Rivest, View planning for automated three-dimensional object reconstruction and inspection, ACM Comput. Surv. 35 (1) (2003) 64–96.
[6] Q. Shi, C. Zhang, N. Xi, J. Xu, Develop feedback robot planning method for 3D surface inspection, in: Proceedings of the 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems in IROS'09, IEEE Press, Piscataway, NJ, USA, 2009, pp. 4381–4386.
[7] J.Q. Gan, E. Oyama, E. Rosales, H. Huosheng, A complete analytical solution to the inverse kinematics of the Pioneer 2 robotic arm, Robotica 23 (2005) 313–336.
[8] B. Kenwright, Inverse kinematics – cyclic coordinate descent (CCD), J. Graphics GPU Game Tools 16 (4) (2012) 177–217.
[9] L.-C.T. Wang, C.C. Chen, A combined optimization method for solving the inverse kinematics problems of mechanical manipulators, Rob. Autom. IEEE Trans. on 7 (4) (1991) 489–499.
[10] P. Chiacchio, B. Siciliano, A closed-loop Jacobian transpose scheme for solving the inverse kinematics of nonredundant and redundant wrists, J. Robot. Syst. 6 (5) (1989) 601–630.
[11] B. Almasri, F. Ben Ouezdou, Human-like motion based on a geometrical inverse kinematics and energetic optimization, in: 2008 IEEE/RSJ International Conference on Intelligent Robots and Systems, September 22–26, 2008, Acropolis Convention Center, Nice, France, 2008, pp. 640–646.
[12] H. Toshani, M. Farrokhi, Real-time inverse kinematics of redundant manipulators using neural networks and quadratic programming: a Lyapunov-based approach, Rob. Auton. Syst. 62 (6) (2014) 766–781.
[13] S. Masayuki, Analytical inverse kinematics for 5-DOF humanoid manipulator under arbitrarily specified unconstrained orientation of end-effector, Robotica, FirstView (2015) 1–21.
[14] D.E. Whitney, Resolved motion rate control of manipulators and human prostheses, Man-Machine Syst. IEEE Trans. on 10 (2) (1969) 47–53.
[15] B. Siciliano, Closed-loop inverse kinematics algorithm for constrained flexible manipulators under gravity, J. Field Rob. 16 (6) (1999) 353–362.
[16] E. Lazarevska, A neuro-fuzzy model of the inverse kinematics of a 4 DOF robotic arm, in: 14th International Conference on Computer Modelling and Simulation, 2012 UKSim, Cambridge, United Kingdom, March 28–30, 2012, 2012, pp. 306–311.
[17] J. Panchanand, B.B. Bibhuti, P.S. Om, Intelligent computation of inverse kinematics of a 5-dof manipulator using MLPNN, in: Advances in Autonomous Robotics Systems – 15th Annual Conference, TAROS 2014, Birmingham, UK, September 1–3, 2014. Proceedings, 2014, pp. 243–250.
[18] H. Wang, Kinematics and control for a personal robot with five degrees of freedom arms. Networking, Sensing and Control, in: 2007 IEEE International Conference on, 2007, pp. 507–512.
[19] B. Siciliano, L. Sciavicco, L. Villani, G. Oriolo, Robotics Modelling, Planning and Control, Springer, London, 2009.
[20] A.M. Lyapunov, The general problem of the stability of motion, Int. J. Control 55 (3) (1992) 531–534.

Please cite this article in press as: A. De Stefano, et al., Modeling a virtual robotic system for automated 3D digitization of cultural heritage artifacts, Journal of Cultural Heritage (2015), http://dx.doi.org/10.1016/j.culher.2015.11.008