Abstract—From the very first hours after disasters such as earthquakes, hurricanes, floods, landslides, and tsunamis, survivors and rescue teams start repairing the devastated vital infrastructure. In doing so, spare parts are frequently needed, which cannot be delivered via destroyed roads or cannot be found in collapsed warehouses. In this work, we present an approach of how spare parts can be reconstructed out of images, and printed out by a normal or 3D printer. Therefore, a structure from motion algorithm is applied to a small number of images, showing the spare part from every direction, which can be captured by any kind of digital camera. Based upon the resulting dense 3D point clouds, a meshed 3D model is computed. After estimating the dimension of the real parts with a straightforward user interface, the meshed virtual model can be scaled correctly. In order to transfer the virtual 3D model back to the real world, our approach provides a 3D model format—for 3D printers—and a stacked paper format—for normal printers. In case no printers are available, the created 3D model can be transmitted via any communication network like GSM, digital radio, or internet to the next available printer or warehouse, nearby. For demonstrating our method, we experimentally reconstruct a gear wheel of a water pump. Finally, we discuss advantages, drawbacks and further steps—necessary for making our approach available.

Index Terms—3D modeling; creation of replica; structure from motion; 3D printing; stacked paper model; rebuilding infrastructure

I. INTRODUCTION AND MOTIVATION

In 2015 98.6 million people were affected by 346 reported natural disasters like earthquakes, heat waves, landslides, floods, storms, and tsunamis [1]. Unfortunately, this seems to form a positive trend, since the number of natural disasters continues to rise year by year [2]. Within our research, we cannot stop this trend or any natural disasters, but we can enable survivors and rescue teams to easily reconstruct broken spare parts for repairing devastated vital infrastructure after these disasters. Therefore we promote computer vision based 3D reconstruction, using simple equipment which might be found in shelters or can be found in everyone’s pocket.

For the reconstruction, the broken parts must be removed from the device or systems to get a 360° view. Around 12 to 36 images of the object are needed, in general, showing it from every direction. This step of capturing photos can be done with any kind of digital camera such as smart phone cameras, photo cameras or webcams. In order to obtain the 3D model, one then starts the automatic 3D point cloud reconstruction, which is based on structure from motion (SfM) algorithms to estimate the 3D structure. Once a point cloud is calculated, the user aligns and scales the virtual 3D model by simple and straightforward user interface (UI) metaphors. Originating from this point cloud, a watertight meshed surface of the part is calculated using a Poisson meshing algorithm. Finally, one can i) print the virtual model directly with a 3D printer to get a real replica, one can ii) print stencils of the model with a normal printer for assembling a sliced stacked model, or on can iii) easily send the 3D model to the next printer available as well as to the next warehouse via any kind of communication networks—since the file size of the virtual model is usually much smaller than of a single photograph.

The main aim of this paper is to propose and evaluate a practicable system of image based reconstruction of parts in disaster scenarios. Further, we resurrect the idea of stacked model creation from 2D materials e.g. printed by normal printers as a temporal solution of fixing the system until the original spare part could be delivered. As the consequence, this paper is organized as follows: In Section II, we describe related work, which forms the basis of our system for image based parts reconstruction. The proposed system is introduced in detail in Section III. This Section is composed of four subsections. The first one gives a complete overview of the system, followed by a discussion of the required equipment and its availability in disaster scenarios. The third subsection introduces and describes the conceptional draft of the UI including its metaphors. Finally, in order to enable everyone to re-implement our approach in the current stage, we describe the prototype implementation in detail. Based on these theoretical considerations, we evaluate the feasibility and assess the quality of the system by experimentally reconstructing a gear wheel of a water pump in Section IV. After discussing the result qualitatively in terms of accuracy, computational time and usability in Section V, we conclude this paper by outlining...
practicability, chances, and drawbacks of our proposed system. Finally, we highlight the necessary improvements to be made – necessary to making our approach available.

II. RELATED WORK

Nowadays a complete research field in computer vision deals with the generation of 3D models from sensor data. Therefore, a vast amount of methods and algorithms [3]–[11] for different kinds of sensors, e.g. time of flight, structure light, and monocular cameras have been developed. Focusing on 3D reconstruction methods, still applicable after disasters, all techniques using special equipment, like depth cameras [12], are left aside in this sections. Thus, it focuses on methods using monocular cameras—these days, build in every smart phone—to reconstruct 3D structures. This exposition is supplemented by a brief overview of approaches for creating watertight meshes based on 3D point clouds.

A. 3D Structures from Images

Photo tourism by Snively et.al. [13] as one famous technique which uses an unstructured collection of photographs of a certain scene, e.g., acquired from the internet, and converts them to a sparse 3D model of the scene. Thus the user can browse, explore and organize large photo collections in a 3D model of the scene. Camera resections, the reconstruction of viewpoints from where the photos are taken, is also possible. Photo tourism reconstructs the model by computing correspondence features of each image using SIFT and RANSAC using descriptors that are robust with respect to variations in pose, scale, and lighting, and runs an optimization to recover the camera parameters and the 3D position of those features. A robust SfM algorithm is the backbone of this automatic system, that estimating 3D structures. Some weaknesses of this system are that it works only with huge collections, above 100 photos. Textureless or repeating structures cannot be reconstructed by this approach, only continuous scenes can be reconstructed as a quite sparse 3D point cloud.

Inspired by photo tourism the software Microsoft Photosynth is a tool for capturing, viewing and sharing your photos in 3D. It works in the same way as photo tourism and automatically reconstructs a collection of overlapping photographs into a 3D model of the space. A scene of 200 photographs is computed in five or ten minutes on an average laptop. To overcome the weaknesses of photo tourism a photography guide is provided which helps the user with how to take photographs so that Photosynth can be used to best advantage. Within this guide many tips or restrictions, like do not shot repetitions, complex occlusions, shiny objects, are given. A plausible collection reconstructs an acceptable model of the scene. Moving or dynamic objects cannot handle by Photosynth [14].

The main purpose of the first two techniques is browsing, exploring and organizing photographic collections in a 3D model. The purpose of Agisoft PhotoScan is the automatic creation of textured 3D models from photographs. To successfully and completely reconstruct a model, photographs shown in a 360° view of the object are necessary. If the software works automatically or interactively depends on the viewpoints, because in the reconstruction process the user must manually outline the object of interest to exclude irrelevant elements like the background or the photographic turntable on each input photograph. Further process steps like aligning photographs, building a dense point cloud, meshing point clouds and creating textured work automatically in PhotoScan. Weaknesses of this software are similar to the previous; also the “solution” is similar, a photograph guideline [15].

Like PhotoScan, Autodesk 123D Catch creates 3D models from a collection of photographs and like PhotoScan it builds a monolithic 3D model. But in 123D Catch the user can manipulate the model in a post-processing phase. To reduce the computational power required by the user’s computer, the overall process is outsourced to cloud computing. The weaknesses, as usual, include no moving objects, no reflections, no under- or overexposure, no blurred photographs, no occlusion etc. [16].

Kowdle et.al. noticed the above mentioned issues of automatic reconstruction for object creation and came up with a semi-automatic approach. In this approach, they put the user in the loop of computational reconstruction, which allows the user to intuitively interact with the algorithm. This application uses an algorithm where the user guides the process of image-based modeling to find a model of the object of interest by interacting with the nodes of the graph. Due to this interaction, the 3D reconstruction achieves a much higher quality in an acceptable time span, especially for scenes with textureless surfaces and structural cues. This semi-automatic approach works with multiple images of a scene, captured from different viewpoints. In the pre-processing steps, a SfM algorithm creates a dense 3D point cloud. This cloud is used for growing 3D superpixel. With user interaction, the superpixels are labeled to segments and finally to the object of interest. With this knowledge, a RANSAC based plan-fitting on the labeled 3D points estimates the 3D Model [17].

In the recent years, a several software solutions like 123D Catch for reconstructing 3D models from multi-view image have been released. In a benchmark [18] based on different unordered collection of photo, we qualitatively and quantitatively compared the resulting 3D models of four software solutions: Agisoft PhotoScan Standard Edition [15], Autodesk 123D Catch [16], VisualSFM [19]–[21] with CMVS [22], and ARC 3D [23]. Therefore particular attention was payed on the ability if the resulting 3D models could be used for printing replicas.

B. Point Cloud Meshing

The use of point clouds increased steadily in the last decades through the introduction of new 3D sensing hardware [24], but they remain rather low-level hardware-related data. With the need to perform high level analysis on 3D objects, the facilitation of advanced point cloud processing algorithms advanced continuously. One important step in processing point clouds is meshing, i.e. the reconstruction of surface meshes.
from point clouds [25]. Since point clouds are often noisy, the creation of hole-free, watertight meshes is often a non-trivial task and requires advanced interpolation algorithms. Poisson meshing, as a well-known technique, creates watertight surfaces from an oriented point set [26], in contrast to e.g. surface triangulation [27] or ball pivoting [28].

III. IMAGE BASED SPARE PART RECONSTRUCTION

Image based reconstruction in controlled, laboratory environments has been proven to work well with accurate results, cf. the Middlebury multi-view stereo benchmark [29], [30]. However, in order to facilitate application of these techniques in natural environments, some obstacles have to be overcome, e.g. handling textureless or shiny surfaces [31] as well as transferring point clouds into CAD-like models [32].

In the following, we propose a system which makes it possible to use image-based multi-view techniques for reconstruction purposes—even with limited equipment. To make this possible, user interaction is an essential and central part of our system for enabling the construction of accurate replicas [17], [32].

A. System Overview

As seen as in Fig. 1, the whole system can be described as straightforward activity diagram comprising steps with only manual activities—marked blue, steps with only computational activities—marked green, and interactive steps—marked yellow. As the starting point, we assume a broken part that should be replaced, but could still serve as a blueprint. Otherwise, a structurally identical part from another system or device must be available. Subsequently, the reconstruction of the needed part is done in six major steps:

I. Within the first step, the user must decide whether or not the part is textureless, transparent or shiny. These issues are the main reasons for potential failures of the image-based multi-view reconstruction process [31]. In cases where the part is textureless, transparent, or shiny, the user must create additional texture without significant modify the 3D structure of the part. In laboratory environments, one can use special equipment like coating spray to prepare shiny or translucent parts [33]. By assuming that non such special equipments are available in the current use case, we experimented with different materials creating a texture-rich surface. In these experiments, we create sufficiently good texture by moisturizing and powdering the part with e.g. dust, sand, or flour, as shown in Fig. 3a. Moisturizing the part beforehand is important to keep the texture fixed when moving the part around to get images from every angle. In case the part already has a texture-rich surface, this step can be skipped.

II. The next preparation step is to place the part on top of a planar but textured surface. This will improve the extraction and mapping of every single camera position, which is important for the triangulation process to compute the 3D point cloud. This texture-rich surface can be a newspaper or a printed shirt. A constraint for the choice of the surface is that repeating patterns should be avoided, since they make it harder for the algorithm to correlate the images to specific view perspectives.

Fig. 1. Activity diagram of the proposed system for reconstructing spare parts in six major steps, where manual activities are marked blue, computational activities are marked green and interactive activities are marked yellow; I, after removing the part from the system or device the user creates additional texture e.g. with dust, flour, or sand if the part is textureless, transparent or shiny; II, the part is placed on top of a textured surface, like a newspaper; user takes 12 to 36 photos with a monocular camera; IV, a smart device or a computer calculates the point cloud, as well as the corresponding watertight mesh; V the user aligns the 3D model according to the X – Y plane and scales the part with easy-to-use UI metaphors; VI, the 3D model can be directly i) printed as replica if a 3D printer is available, in normal cases where no 3D printer is available the 3D model can be ii) exported as stencils which can be assembled to a sliced replica, or iii) the 3D model can be transmitted via any communication network like GSM, digital radio, or internet to the next warehouse or printer.
III Next up, the use of digital equipment is required. With a monocular camera, like a standard photo camera, the user takes a small number of about 12 to 36 photos, showing the part from 360° including all important details.

IV Using the captured images, a SfM algorithm computes a dense point cloud of the part. Due to the fact, that point clouds usually contain points of the background, the user must support the smart device, tablet or a computer by recognizing the background pixels. This can be done e.g. by using a cutting plane or color segmentation. As soon as a clean point cloud is available, it can be meshed to get its surface. In optimal cases, the resulting mesh is watertight, i.e. the resulting surface has no holes or missing elements.

V The fifth step is the alignment of the 3D model according the X – Y plane and the determination of the scaling factor. Therefore the user rotates the model on the screen until the Z axis is perpendicular to the screen. Once the virtual model is in place, the model will be aligned by pressing the set align button, cf. Fig. 2(d). Due to the fact, that cameras will mostly be uncalibrated, the 3D model has a random scaling factor. For estimating the correct scaling factor, the user places the part on the screen and scales the 3D model until the real part as well as the 3D model are coherent. The factor is set after pressing the set scale button, as shown in Fig. 2(e)–(f).

VI The final step is the translation of the virtual 3D model back to one or several real world replicas. For this intent, the user has three different options, depending on the available equipment. In the unlikely case, i) a 3D printer is available, the created 3D model can directly be printed, cf. green replica in the middle of Fig. 4. This is the fastest way to get a replica, but only possible with the usage of a 3D printer. Because 3D printers are normally not available in shelters, the user can use ii) a normal printer to get stencils, as shown as in Fig. 3(d). Therefore the thickness of available material like cardboard or plywood must be provided. The system then calculates the number of needed stencils, as well as their shapes. After printing the stencils, the user fixes them on the available material and cuts the material along the stencils’ contours. Finally, the user assembles all parts to a sliced stacked 3D model. An example stacked replica is given on the left hand side of Fig. 4. In case other production machines, such as CNC machines, are available, the stencils could also be used to create a stacked 3D model from them. Nevertheless, if no printing device is available, the created 3D model can iii) be transmitted via communication networks like GSM, digital radio, or internet to the next printer or a warehouse nearby. Due to the fact that the created 3D model contains only information about the part and does not contain any background information, the amount of data which must be transmitted via communication networks is reduced to a minimum. Accordingly, it is possible to transmit these 3D models via communication devices with a low bandwidth.

Lastly, it should be noted that our system focuses on enabling image-based spare part reconstruction after disasters with user-interaction and does not yet provide a fully automated reconstruction system.

B. Necessary Equipment

With the intended use case of reconstructing spare parts after disaster scenarios, we try to minimize the number of needed equipment. These conditions are already considered in the design of the activity diagram of Fig. 1. Therefore the minimal necessary equipment for each of the process steps is:

I: textured material like dust, sand or flour (if the object is textureless)
II: textured surface
III: monocular camera (like photo camera, smart phone camera, webcam or tablet camera)
IV/V: computer (like laptop, workstation, tablet PC or smart phone)
VI: printer (like 3D printer, printer or CNC machine) or any communication network (if no printer is available)

The required equipment for I and II should be easy to access, even in areas with destroyed infrastructure. Phases III, IV and V could be more problematic, although smart devices are nowadays in widespread use, even in third world countries. A printer for step VI might be unavailable, but communication is usually set up again quickly in order to coordinate disaster relief missions. Steps III to VI require in principle also electricity, which can e.g. be generated by an emergency generator.

In summary, the requirements for the system should be low enough in order for it to be applicable in many disaster scenarios. Still, there might be some cases, where e.g. electricity is not available in some period after the event and where our approach is, therefore, impractical at the beginning.

C. User interface

In current software development efforts, the UI is, in particular, important for the acceptance of a tool. Thus, even for research projects, it is essential to consider possible implementations for a thoughtful UI design, with the aim of creating a usable software system.

After starting the 3D reconstruction software or application, the user should be guided through the straightforward process of our system. First, information on how the user has to prepare the part is provided, as intended in steps I and II. This information should be provided in both a textual and pictorial form.
After the user finished the preparation steps, the process of taking photos can be started. Therefore the user starts the photographing process by clicking the button *take photo*, as visualized in Fig. 2(a). Note, only the most important screens of the current UI implementation are illustrated in Fig. 2. If the user takes a photo, it is displayed on the screen as shown as in Fig. 2(b), until the button *take next* is pressed. When *take next* is pressed, the user is shown the standard photographing UI of the device. In this important step of taking the photos of the object, there is still much room for improvement of our preliminary UI. One approach is e.g. a user guidance, which shows for every photo the percentage of overlap to the previous one. With such an indicator, the process could be optimized in taking the most appropriate number of photos that are also best suited for the reconstruction process.

If a sufficient amount of photos is taken, the step IV, the reconstruction of the 3D model is started by pressing *calculate point cloud*, seen at Fig. 2(b). Due to the fact that the creation of the point cloud could take hours on a low performance device, a processing indicator is shown, which gives the user feedback that the process is still running.

Once the point cloud is calculated and meshed, is visualized to the user, as seen in Fig. 2(c). With one-finger touch-gestures, the point cloud can be rotated with respect to a rotation center. The rotation center can be changed by double-tapping on a certain point of the screen. The normal two-finger zoom-in and zoom-out gestures scale the point cloud. To align the 3D point cloud the user rotates the 3D model until the Z axis is perpendicular with respect to the screen, which spans the X−Y plane, cf. Fig. 2(d). By placing the part on the screen, the user scales the 3D model until the 3D model covers the original model. As seen in Fig. 2(e)–(f), this should be done from at least two viewpoint perspectives in order to assign the correct scaling factor to the 3D model. If the part’s dimensions exceed the screen size, the scaling process works with partial elements of the part as well.

The UI for the last step VI is rather simple. In order to print the model i) directly with a 3D printer, a normal printing dialog is used. For printing ii) the stencils, a normal printing dialog is altered with the attribute of the material thickness, used for assembling the stacked 3D model. For enabling the transfer of the 3D model, it can be stored using a normal file save as dialog.

D. System Implementation

The prototype of the proposed system is implemented on a Windows tablet. Therefore a user interface in HTML5 on a local host was implemented. HTML5 also offers the possibility of accessing input devices, like webcams. Based on this we designed a UI, where the user can take photos...
from the object, according to step III and store them into a temporal folder. Because the part preparation steps I and II create ideal conditions for the SfM algorithm, the 3D point cloud reconstruction can be done with standard reconstruction software like VisualSFM [19] or Agisoft PhotoScan [15]. In this prototype, a local batch file is called from the UI via JavaScript. This batch file executes the freely available software VisualSFM, which creates a point cloud out of the taken photos. The resulting point cloud is then visualized using HTML5 with WebGL to the user. For creating a watertight mesh, another batch file is executed, but this time, the batch file calls the Poisson meshing by Kazhdan and Hoppe [26]. The realizations of the alignment process to the $X-Y$ plane and the scaling process are done using WebGL again as illustrated in Fig. 2(d)–(f). The aligned, scaled and meshed 3D model is now stored in the ply data format.

Hence, the 3D model, stored in the *.ply format, can be sent directly to i) 3D printer engines or ii) via a communication network to other location. The stencils for the stacked 3D model are created with a MATLAB script directly out of the ply data. As parameters for this script, the thickness of the material must be set, which influences the number of stencils and the distribution of the slicing planes for contour extraction.

IV. EXPERIMENTAL EVALUATION

In order to show that our system can already be successfully applied, we tested it with the experimental reconstruction of a gear wheel out of a water pump. As intended by step I the gear wheel is first removed out of the pump, shown in Fig. 4(c). On the left hand side of Fig. 4 one can see the original gear wheel. Since the black gear is textureless, it is moisturized and powered with dust for creating additional feature points to increase the amount of points in the point cloud. After placing the gear on top of a newspaper, 36 images were taken with the build in tablet camera—four of these photos can be seen in Fig. 3(a).

By removing background pixels out of the created point cloud in step IV, one recognizes that the applied SfM algorithm—like any other algorithms—is not able to calculate points of the bottom side of the part. Without these points, the used Poisson algorithm computes multiple coherently meshed models and not a single coherently meshed part as needed. Since the gear wheel has symmetries, the points from the top side are reflected to the side were fewer points exists. Unfortunately, we were not able to do this on the tablet, but instead—to be able to show that the overall system works—we mirrored the points of the 3D model using MeshLab [34]. In Fig. 3(b) the resulting point cloud is illustrated. Afterwards, the Poisson algorithm creates a watertight surface 3D model, which is shown in Fig. 3(c).

As shown in Fig. 2(d)–(f), the virtual 3D model is aligned and scaled using our simple UI metaphors. For the qualitative evaluation of our reconstruction, the 3D model is printed in green polylactide (PLA) plastic, as seen in the middle of Fig. 4. To test the slice stacked model, we choose a $2\text{mm}$ thick cardboard as material—this results, in our case, in six stencils. These stencils were printed on a normal paper sheet, which is afterwards glued on the cardboard. The cut out slices are now assembled to a replica, seen as white replica on the left side of Fig. 4.

In Fig. 4(c) one can see the original, as well as the repaired water pump with the use of the green 3D printed and the white stacked replica. The footprint of the final meshed 3D model is in our case only $700kB$, which makes it ideal for transmission at very low bandwidths.

V. DISCUSSION

The experimental results suggest that the accurate reconstruction of spare parts with image-based 3D reconstruction including the creation of replica with only limited equipment, e.g. after disasters, is possible. Take Fig. 4 into account for qualitative results. The complete reconstruction process without the printing of the replica takes about four hours on the tablet, where the computational power of the tablet seems to be a bottleneck for running VisualSFM as well as for the Poisson algorithm. The creation of the stacked sliced model including cutting and assembling takes two hours. One issue at the present stage of development is the lack of UI metaphors for touch devices that enable the user to improve the density of the point cloud by indicating symmetries of the object. Mirroring the point cloud over the UI could lead to significant improvements in the point cloud density.

For the demonstration of the accuracy of our system, we applied it on a relatively small gear wheel. The success in assembling the final replica into the original system shows its accuracy. Further our system seems to be highly scalable in terms of its accuracy and only limited by the material attributes used for creating the replicas.

©2016 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.
©2016 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

Of course, this process is rather pointless for very common parts that are stored in warehouses or where 3D models are already available from their part number or other identifiers. Thus our system rather concentrates on the reconstruction of unique parts which are not easily available by any other means.

VI. CONCLUSION

In this paper, we present a scalable system for image-based 3D reconstruction of spare parts after disasters. By reconstructing a gear wheel of a water pump, we experimentally evaluated our approach. Within this evaluation, we recognized that our system must be mainly improved in two aspects. First, new UI metaphors for touch devices must be introduced that enable the user to apply simple symmetric operations to increase the density of the point cloud. Second, the processing time must be drastically reduced.

Next to these two aspects, the implementation of a guided photography UI, similar to the augmented reality approach of ProFORMA [9] would be an improvement. This could also help to reduce the processing time, due to lowering the number of photos. In future user experiments, non-expert users should use our system to evaluate the usability of our UI and its metaphors. Next to the usability evaluation we are planning to benchmark our interactive approach of spare part reconstruction with the ordinary, completely manual creation of spare parts, with special interest on the time needed for creating replicas, the utilized equipment, and the used materials. This benchmark is very important because some colleagues already argue, that they are able to reconstruct possible needed parts faster and more accurate without using any reconstruction system.

A quite general open research question is, how an algorithm can recognize whether or not a certain point in a point cloud belongs to the part or to the background. Developing such a segmentation algorithm, which shows a good performance, would further improve the resulting 3D model. Beyond that, a new research area must be introduced with the aim to construct (temporary) spare parts from images without the need of disassembling parts, identifying broken parts, or even having the broken part at hand. As shown in Fig. 5, this new research area expands the current process with the recognition of parts, detection of broken parts, and understanding of the system including its parts to reconstruct 3D spare parts from multiple images or video sequences.

In the long term future, we are trying to arrange a reconstruction first aid kit for less than 200$ which can be made available in shelters or for first respond rescue units. It should include all necessary equipment like a tablet PC, an accu-pack and a mobile 3D printer, which some companies like OLO 3D are developing right now [35]. This kit should allow for the reconstruction of parts of 120mm $\times$ 65mm in less than two hours.

Fig. 4. Top view (a), side view (b), and the assembled view (c) of the resulting parts created during the experimental evaluation of the proposed system. On the left hand side, in black the original part; in the middle in green the reconstructed 3D printed replica; on the right hand side in white the sliced stacked 3D model.

Fig. 5. Intended reconstruction process with the aim to construct (temporary) spare parts from images without the need of disassembling parts, identifying broken parts, or even having the broken part at hand; it expands the current process, red dotted arrows, with the recognition of parts, detection of broken parts, and understanding of the system including its parts to reconstruct 3D spare parts from multiple images or video sequences.
REFERENCES


