

FIRST EXPERIENCES WITH KINECT V2 SENSOR FOR CLOSE RANGE 3D MODELLING

E. Lachat, H. Macher, M.-A. Mittet, T. Landes, P. Grussenmeyer

Photogrammetry and Geomatics Group, ICube Laboratory UMR 7357, INSA Strasbourg, France
(elise.lachat, helene.macher, marie-anne.mittet, tania.landes, pierre.grussenmeyer)@insa-strasbourg.fr

Commission V, WG V/4

KEY WORDS: Kinect v2 Sensor, RGB-D camera, Range Imaging, Calibration, Point Cloud, Assessment

ABSTRACT:

RGB-D cameras, also known as range imaging cameras, are a recent generation of sensors. As they are suitable for measuring distances to objects at high frame rate, such sensors are increasingly used for 3D acquisitions, and more generally for applications in robotics or computer vision. This kind of sensors became popular especially since the Kinect v1 (Microsoft) arrived on the market in November 2010. In July 2014, Windows has released a new sensor, the Kinect for Windows v2 sensor, based on another technology as its first device. However, due to its initial development for video games, the quality assessment of this new device for 3D modelling represents a major investigation axis. In this paper first experiences with Kinect v2 sensor are related, and the ability of close range 3D modelling is investigated. For this purpose, error sources on output data as well as a calibration approach are presented.

1. INTRODUCTION

Capturing three dimensional environments and objects is a necessary task for many applications in different fields. This is nowadays widely made possible thanks to various technologies such as laser scanners, stereo-vision or triangulation-based systems for example. Regarding some aspects like price or computation time, the recent RGB-D cameras offer new possibilities for the modelling of complex structures particularly in cultural heritage documentation.

This type of sensors presents several benefits demanded initially by the gaming and entertainment industry. They are easy to use and some of them are low-cost compared with laser scanners. Regarding these advantages, their use was extended to many other areas such as 3D scene reconstructions or pattern recognition (Kolb et al., 2009) as well as in the robotic field (May et al., 2006).

Kinect v1 and Kinect v2 are based on two different imaging principles. As the accuracy of Kinect v1 limits its use for some engineering measurement tasks, Kinect v2 will probably give better results considering the new technology.

The aim of this paper is first to present the new sensor and its functionalities. In a second part, several tests are carried out and a calibration approach is proposed in order to improve output data. This will highlight not only the errors related to environment and captured object but also errors related to the system itself. Finally, the benefits of the new sensor are listed.

2. KINECT V2 SENSOR PRESENTATION

2.1 Characteristics

Even though Kinect for Windows v2 sensor relies on a different technology than Kinect v1, it also allows the acquisition of

three different output streams. It is composed of two cameras, namely a RGB and an infrared (IR) camera, from which just one can be seen on the left side of the sensor (Figure 1). The active illumination of the observed scene is insured by three IR projectors.

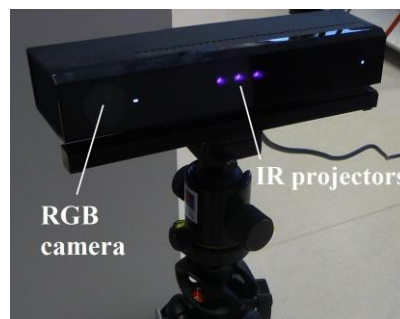


Figure 1. Kinect v2 sensor on a photographic tripod

The RGB camera captures color information with a resolution of 1920x1080 pixels, whereas the IR camera is used for the real-time acquisition of depthmaps and also IR data with a 512x424 pixels resolution. The whole acquisitions can be carried out with a framerate up to 30 Hz. A last feature to be mentioned is the field of view for depth sensing of 70 degrees horizontally and 60 degrees vertically.

The technical specifications provided by Microsoft announce an operative measurement range from 0,5 m to 4,5 m. This last characteristic will be verified through different experiments presented in this paper.

To enable the use of the sensor for developers and researchers, the official Microsoft SDK 2.0 (Software Development Kit) is free downloadable. It provides not only the drivers, but also a set of functions or code samples that can be used for own implementations. One should also note the introduction of the

Kinect Fusion tool in this SDK, which was initially released by Microsoft for the Kinect v1 because of the success of its device. This tool enables thanks to ICP algorithms the direct creation of 3D meshes under different formats (.stl, .obj or .ply) by using the Kinect as a scanning device moved slowly around the object.

2.2 Distance Measurement Principle

As mentioned on Microsoft website and on a Canesta patent, the Kinect v2 sensor is based on time-of-flight principle, whereas the previous Kinect device uses structured light to reconstruct the third dimension. Even though time-of-flight range imaging is a recent technology, many books deal with its principles and its applications (Grzegorzek et al., 2013, Hansard et al., 2012, Kolb and Koch, 2009, Remondino and Stoppa, 2013). Moreover in the last decade, a few companies developed range imaging cameras based on this measurement principle, which can be direct or indirect. The distinction between these two different measurement types is based on the light source used, either light pulses (direct ToF) or amplitude modulated light (indirect ToF). Nevertheless, the basic principle stays unchanged: knowing the speed of light, the distance to be measured is proportional to the time needed by the active illumination source to travel from emitter to target.

As Canesta made use of indirect time-of-flight before Kinect v2 sensor development, we can assume that this methodology also appears in the new sensor investigated here. Indeed, direct time-of-flight requires very costly clocks to accurately determine the runtime of single pulses. Therefore it seems unlikely that such a component be part of a low-cost sensor. For indirect time-of-flight systems based on light modulation, a phase shift between emitted and received signal is measured, replacing the direct measurement of runtime. In that case, the estimated distance between sensor and captured object depends on the determined phase shift $\Delta\phi$ by equation (1) (Kolb and Koch, 2009).

$$d = \frac{\Delta\phi}{4\pi f} \cdot c \quad (1)$$

where f = modulation frequency
 c = speed of light

It should be noticed that, unlike other time-of-flight sensors, it is impossible to act on the modulation frequency or on the integration time of the input parameters with Kinect v2 sensor.

2.3 Point Cloud Acquisition

Sensors based on the digital imaging technology introduced above deliver a measurement of the distances between an entire scene and the sensor. These distances values are directly stored in a matrix which size corresponds to the depth sensor resolution. Indeed, for each pixel of the 512x424 depth images, the measuring device estimates in real-time a distance value to the corresponding object point. This output data is the so-called depthmap (see an example Figure 2), a two-dimensional image from which the 3D-coordinates of the scene can be calculated. After a few post-processing steps, it is then possible to obtain indirectly point clouds of the captured scene or object.

Assuming that Kinect sensor uses a standard lens, camera intrinsic parameters such as focal length f and principal point coordinates (c_x , c_y) can be determined. The Z-coordinate for a pixel is already known because it corresponds to the value

stored in the depthmap. Using these parameters and the perspective projection relationship (see Dal Mutto et al., 2013), a 3D point X in the camera coordinate system can be mapped from the homogenous image coordinates of a pixel $p = [u, v, 1]^T$ through equation (2). As each pixel of the depthmap represents a value for the corresponding point cloud, the calculated point clouds count a number of 217088 points.

$$\lambda p = KX = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \Leftrightarrow \begin{cases} X = \frac{u - c_x}{f_x} \cdot Z \\ Y = \frac{v - c_y}{f_y} \cdot Z \end{cases} \quad (2)$$

Because of the recent release of the Kinect v2 sensor, so far only a little amount of algorithms allows an off-the-shelf use of the sensor. However, the Microsoft SDK (Software Development Kit) provides multiple functions for the recording of the different data types. Respecting the fact that we don't control the mapping functions used by Microsoft to generate point clouds from depthmaps, we developed our own acquisition protocol. Implemented in C# through some code samples available in the SDK, this program enables us to acquire timed acquisitions for each of the different possible output data (RGB images, infrared images, depthmaps, and colorized point clouds).

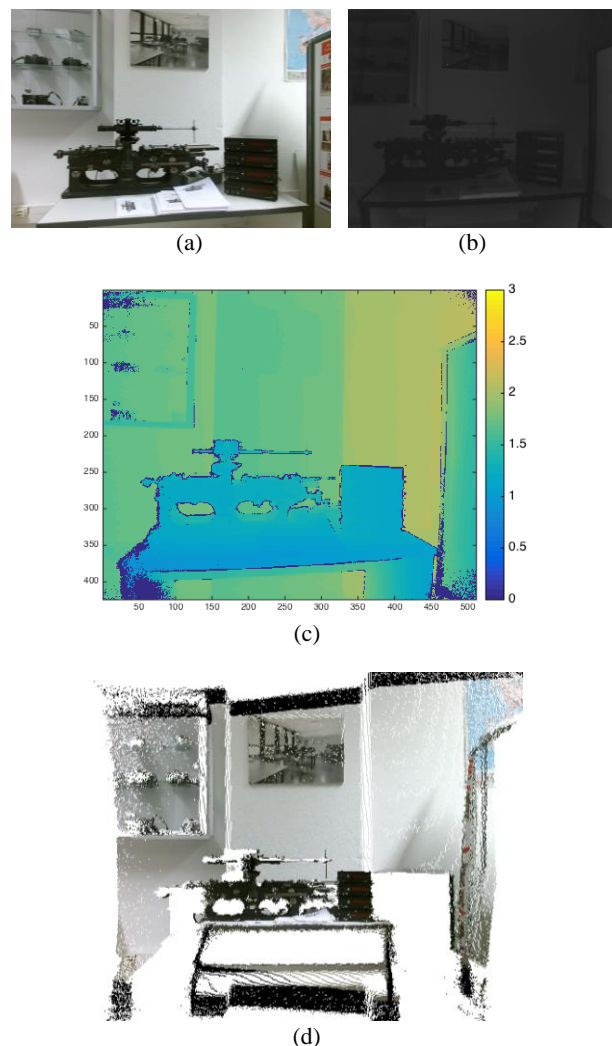


Figure 2. Output data: (a) RGB image; (b) infrared image; (c) colored depthmap (colorbar in meters); (d) colorized point cloud

3. INVESTIGATION OF SYSTEM PROPERTIES

A good knowledge of sources of errors affecting the measurements of a system is needed to quantify the accuracy of the data provided by it. Even if a large number of these sources are interdependent, we can however try to classify them into four groups, as it is done for terrestrial laser scanners (Reshetyuk, 2009): instrumental errors, errors related to the scanned object, environmental errors and methodological errors. Instrumental errors are usually systematic errors assigned to the system design. They should potentially be removed by system design improvement or by calibration. The environment in which the acquisitions are performed has an influence (temperature, brightness, humidity, etc.), as well as the characteristics of the observed object (e.g. albedo, reflectivity, geometry). Once the camera is on its tripod, the user intervention is limited to choosing settings (e.g. time interval, types of output data), therefore the automation level seems to be relatively high. However, errors can occur according to the distance-to-object chosen for the acquisition, the incidence angle, etc. The point cloud quality depends also strongly on the algorithms used for creating the output data (mapping functions, calibration parameters...). It is essential to determine the influence of these parameters on the quality of the distance measurement. To do that, a few tests are presented in this section.

3.1 Influence of Frames Averaging

Since individual frame acquisitions suffer from noise inherent to the sensor and its technology, averaging successive frames is a possible improvement step to overcome this phenomenon. This part deals with the influence of the number of acquisitions on the precision of the final measurement. Considering a static scene, several repeated measurements with 500 ms interval are carried out from one unique camera location. A statistical analysis over a large sample of 500 acquired depthmaps has been performed. We also analyzed the behavior observed with 20, 50, 100 and 200 successive depthmaps.

First, acquisitions have been made from an office scene which mainly contains objects of squared shape (Figure 3a). Using the acquired depthmaps, for each pixel the mean value and the standard deviation of distance measurements were calculated. For this non planar scene, the computed standard deviations shown in Figure 3b are particularly wide at the objects boundaries, but also on reflective surfaces such as metallic parts. This phenomenon can be explained by the measurement noise which appears in these specific cases. It can also be associated to speckle noise present in infrared data. Moreover, considering the different sizes of samples, one should notice a low impact on the computed standard deviations from the 50 depthmaps. Indeed, for 20 successive frames the result seems to be less smooth than for the other samples sizes. However the effect observed at objects borders does not change a lot between the 50 and 500 samples. While frames are added, the main differences arise at far ranges (from about 3 meters), where depth inhomogeneities are reduced and the visual results smoothed. We will see in section 4 that far range measurements are not reliable. For this reason, they will not be treated and considering 50 successive depthmaps seems to be enough.

Secondly, the same test was repeated by positioning the Kinect device in front of a planar wall, at a distance of approximately 1,15 m. Histograms of the distance measurements performed for a central area of one pixel and its neighbors (3 x 3 pixels) were

realized for the different samples mentioned before. Figure 4 shows the result obtained with 100 successive measurements. In that case, the standard deviation is lower than 1 mm (approximately 0,7 mm). It can be noticed that this value is almost constant for all sizes of samples we considered. In conclusion, as for the experiment presented above, a larger sample does not provide a better precision. The future tests will thus be carried out with a smaller amount of acquisitions.



(a)



(b)

Figure 3. (a) Acquired scene; (b) Standard deviations calculated for each pixel of 100 successive depthmaps (colorbar in cm)

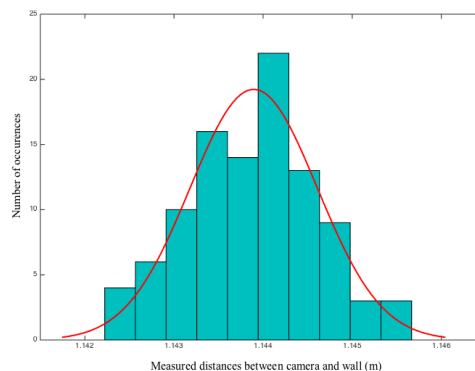


Figure 4. Histogram of 100 distance measurements performed for a 3 x 3 pixels central area

3.2 Pre-heating Time

Previous studies have shown that some RGB-D sensors need a pre-heating time before providing reliable range measurements (Mittet et al., 2013). For checking this phenomenon, the time delay necessary for Kinect v2 to achieve constant measurements has been determined. The sensor was placed parallel to a white planar wall and measurements were carried out each 15 seconds during one and a half hour (360 depthmaps recorded). For each recorded depthmap, the distance from the sensor to the wall was determined in a central area of ten per ten pixels. This area was chosen according to the maximal intensities in the corresponding infrared data.

This test permits to establish the value of distance measurements as a function of time. Figure 5 shows that the distance varies from 5 mm up to 30 minutes and becomes then almost constant (more or less 1 mm). It can be noticed that the sensor's ventilator starts after 20 minutes. For future tests, the pre-heating time will be respected even if the distance variation is quite low.

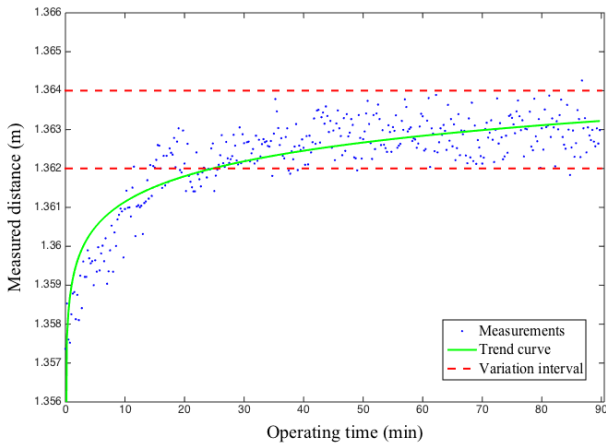


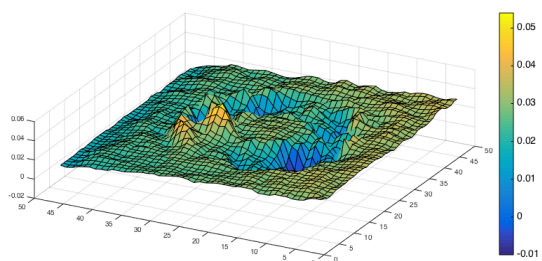
Figure 5. Value of distance measurements as a function of time

3.3 Influence of Materials and Colors

To assess the effects of different materials on intensity as well as on distance measurements, samples characterized by different albedos and roughness were measured. The sensor was placed parallel to the plane regrouping the samples (Figure 6a).



(a)



(b)

Figure 6. (a) Samples characterized by different albedos and roughness; (b) zoom on the red frame: depth variations (m) observed on this area including a compact disk

As shown in Figure 6b and Figure 7, reflective and dark materials stand out among all the samples. Indeed, the intensity provided for very reflective as well as very dark surfaces is

particularly low. Thus, the corresponding distances in the depthmaps are larger than expected. This effect in the depthmap is mainly important for a compact disk, i.e. high reflective material as illustrated in Figure 6b. For this sample, distances vary up to 6 cm. It is also visible in Figure 7, which presents a checkerboard of grey levels and the corresponding intensity image. The intensity increases with the brightness of the object (except when it is also very reflective or transparent). On the white parts of this checkerboard, a speckle phenomenon appears and illustrates the measurement noise.

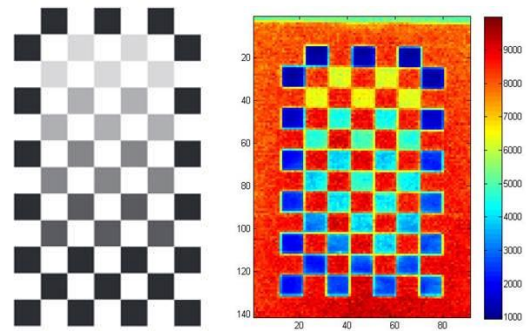


Figure 7. Checkerboard of grey levels (left) and corresponding intensity image (right)

Concerning the level of roughness of the grainy textures on the test panel in Figure 6a, their influence is not clearly visible because of the noise inherent to the point cloud.

3.4 Outdoor Efficiency

As the previous Kinect was not adapted for sunny outdoor acquisitions, the influence of the brightness conditions on the measurements was also studied. Outdoor acquisitions were performed during a sunny day.

This experiment shows that the sensor is able to work during a sunny day provided that the light does not directly illuminate the sensor. Indeed, strong backlighting conditions cause sensor's disconnections from the computer.

Considering the quality of the acquisitions carried out during a sunny day, two phenomena appear: the number of "flying pixels" increases particularly on the edges of the sensor field of view and the number of points decreases with the light intensity. The "flying pixels" effect is shown Figure 8.

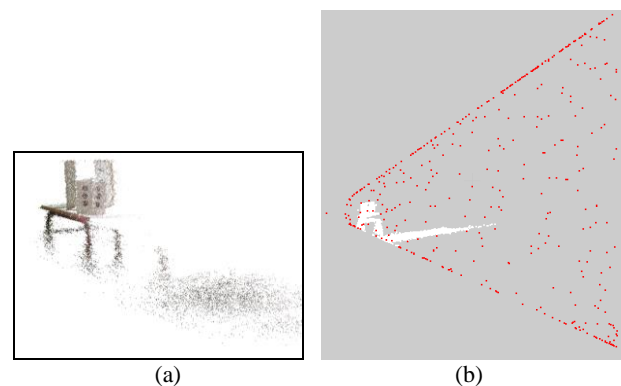


Figure 8. Point cloud acquired in sunny conditions: (a) scene without "flying pixels"; (b) scene (white) and "flying pixels" (red) in a profile view

4. GEOMETRIC AND DEPTH CALIBRATION

As the lenses used in time-of-flight cameras are standard, the acquired data suffer from distortion effects that can be estimated and corrected through successive calibration steps. Since we hope to use the Kinect v2 sensor as a measuring device, calibration should enable the acquisition of reliable 3D metric information from the acquired point clouds. Because of the specificity of range imaging cameras, the distinction between two successive approaches has to be made.

4.1 Geometric Calibration

It appears in lots of works e.g. (Hansard et al., 2014) that time-of-flight cameras can be geometrically calibrated with standard methods. For this purpose, as for common 2D sensors, several images of a planar checkerboard were taken under different points of view. It is worth noting that the infrared data were used to handle this geometric calibration, because depth and infrared output streams result from the same sensor.

To determine the necessary intrinsic parameters, our dataset was treated by a Camera Calibration Toolbox proposed by (Bouguet, 2013) under the Matlab software. This algorithm is mainly based on well-known camera models. However, one should underline the fact that changing or removing some of the images makes the computed results vary from a few pixels. This phenomenon is only due to the low sensor resolution. As a matter of fact, best calibration results are considered regarding the lower uncertainties on the parameters.

Note that the Microsoft SDK mentioned earlier also provides a function that returns all these intrinsic parameters (namely *GetDepthCameraIntrinsics()* function). Deviations towards self-computed results are shown in Table 9.

| | Self-computed calibration | | SDK function | Deviations |
|--|---------------------------|---------|-----------------|-------------------|
| | Values (pixels) | Std. | Values (pixels) | (absolute values) |
| Focal length (x) | 364,7 | 1,6 | 366,0 | 1,3 |
| Focal length (y) | 366,1 | 1,5 | 366,0 | 0,1 |
| Principal point (x) | 255,8 | 2,4 | 258,6 | 2,8 |
| Principal point (y) | 203,7 | 3,1 | 206,5 | 2,8 |
| Distortion parameters (radial, then tangential) | | | | |
| K1 | 0,08708 | 0,01976 | 0,09357 | 0,00649 |
| K2 | -0,16515 | 0,10795 | -0,27394 | 0,10879 |
| K3 | -0,00321 | 0,00365 | 0,09288 | 0,09609 |
| P1 | -0,00345 | 0,00286 | n.a. | |
| P2 | 0 | 0 | n.a. | |

Table 9. Estimated intrinsic parameters using self-calibration, with Microsoft function, and related deviations

The application of these distortion coefficients directly on our point clouds computation algorithm allows the gaining of calibrated three dimensional dataset, which is no more affected by the actual distortions of the initial depthmap. It can also be noticed that the point clouds obtained by applying directly the Microsoft mapping functions seem to be pre-calibrated at least for radial distortions of the lens. Indeed, considering acquisitions of a planar surface, the results are visually similar between a point cloud computed through our method with corrections and a point cloud directly obtained with mapping functions. This phenomenon appears especially on the corners of the captured plane, which are not right angles but rather distorted. As a matter of fact, the intrinsic parameters stored in

the device are certainly considered in the point clouds mapping functions.

Moreover, a geometric calibration could also be processed for the RGB sensor, which uses a different lens with a different resolution. However, as we do not need the colorimetric information for the rest of the experiments, this calibration has not yet been performed.

4.2 Depth Calibration

A systematic depth-related deformation, also depicted as wiggling error by some authors (e.g. Lindner et al., 2010) can be observed while working with time-of-flight devices. This deformation is due in part to inhomogeneities in modulation process. Lots of works have been devoted to the understanding of error sources in order to minimize their effects. As a matter of fact, calibration of time-of-flight sensors was a major issue in many studies in the last decade. Most of the time, the introduced approaches are based either on Look-Up Tables (Kahlmann et al., 2006) or on curve approximations with B-splines (Lindner et al., 2010). In both cases, the aim is the storage of depth errors depending on the measured distance.

A common way to assess the distance inhomogeneity consists on positioning the camera parallel to a white planar wall at different well-known distances. In our study, the wall has been surveyed beforehand with a laser scanner (FARO Focus3D) in order to assess his planarity with a device of assumed higher accuracy as the investigated sensor. After that, a line has been implanted by tachometry perpendicularly to the wall. Marks have then been fixed on this line at predetermined distances, from which the acquisitions were made with the Kinect placed on a static tripod, as shown in Figure 10.



Figure 10. Depth calibration experiment (observed area in dotted lines)

The marks were implanted at ranges from 1m up to 6m from the reference wall, with 25 cm steps to progressively move the sensor away from this wall. An additional position at 0,8m was also set up, even though Microsoft announces a minimal range of 0,5 m. Indeed, measurements acquired at a range of 0,8 m already present a high standard deviation compared to other ranges, as can be seen on Figure 11. This phenomenon will be more important for measurements at a closer range. Moreover to limit the influence of noise, at each sensor position 50 successive depthmaps of the wall were acquired with intervals

of one second. Finally, to insure the parallelism between sensor and wall and thus reduce the possible rotation effect, the distances were also surveyed by tachometry at each new position, using little prisms at the two sensor extremities.

Inspired by the work of (Lindner & Kolb, 2006), a two steps distance calibration based on B-splines approximation was carried out. Firstly, the accuracy of distance measurement for a small central matrix of 10x10 pixels has been investigated. Secondly, the deformations extended to the whole sensor array have been assessed, in order to highlight the need of processing a potential per-pixel correction.

4.3 Results

First aim of this experiment was to determine the deviations between real and measured distances as a function of the depth, to produce the result presented in Figure 11. As depicted in this graph, distortions of the averaged central area vary from about 0,5 cm to 2,7 cm. Knowing the curve parameters allows the correction of the measured distances. Another interesting information arises from the observation of the standard deviations calculated over the 50 depthmaps acquired per station. As it can be seen, the standard deviation increases with the range, which means that the measurements become more scattered around the estimated distance when the sensor moves away from the scene. Moreover, at the nearest range of 0,8 m acquired in this test, the standard deviation is higher than for all other sensor positions. As a matter of fact, measurements realized at the minimal announced distance of 0,5 m would probably be still less reliable. Secondly, an important variation of about 1,5 cm at range 4,5 m is observed. This observation confirms the upper value for distance measurement reliability given by Microsoft. The future acquisitions will be limited to this suitable range, observing that in this case the distance deviations are reduced to an interval of about 1 cm.

Looking at this graph, it can also be noticed that despite satisfying results all deviations are positive, meaning that the sensor would systematically measure too short distances. It seems that a systematic offset occurs because of the constant distance (approximately 2 cm) between fixing screw and lens. Indeed, the Kinect was accurately placed on the tripod with respect to its fixing screw, which definitely does not correspond to the optical center of the lens.

In order to eliminate this systematic effect, we suggest subtracting the values of the nearest range (0,8 m) to the values of each respective position. Thus, observing calculated intervals with respect to a first station also affected by the systematic offset provides values no more depending on the unknown misalignment. The realized graph presents the same trend curve as Figure 11, but with values varying between about 5 and -5 mm for ranges from 0,8 to 4,5 m.

Finally, plotting the measured distances over the real distances provides a very high correlation coefficient of $r = 0,999$. According to the measurement task, it is possible to neglect the depth calibration step.

The second part of depth calibration consists on an extension of the analysis to the whole sensor array. For this purpose, for datasets of the planar wall acquired before at the ten first positions (from 0,80 m to 3 m), fitting planes were computed by means of least-squares adjustment. To fit correct planes in the Kinect point clouds, some outlying points corresponding to null values in the depthmaps and thus very far from the wall were not considered. Then, the Euclidean distances between a point cloud and its corresponding plane are calculated. It provides a residuals matrix which represents for each pixel of the considered depthmap the deviation between approximating plane and actual measured distance. This representation is shown in Figure 12a for a distance of 1,25 meters with deviations in millimeters. In addition, the histogram of the residuals obtained for this range is also presented (Figure 12b). It appears that the values are normally distributed and almost centered on the null value, with a standard deviation of about 4 mm. It can be mentioned that the standard deviations calculated for the fitted planes increase with the range from about 3 mm up to 16 mm for the latest range of 3 meters.

Moreover, as it can be seen in Figure 12a, the deviations with respect to the adjusted plane are more important at image boundaries and especially on the corners. As known from the literature, these radial effects increase with the distance to the wall. In our case, they can reach variations of a few centimeters up to tens of centimeters in the corners at 3 meters range.

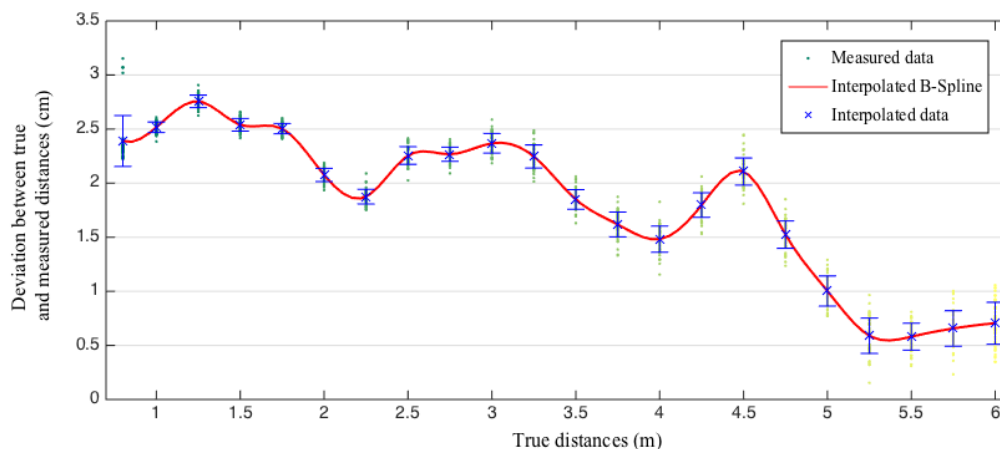


Figure 11. Deviations of measured distances with respect to true distances (without correction of offset distance between fixing crew and lens)

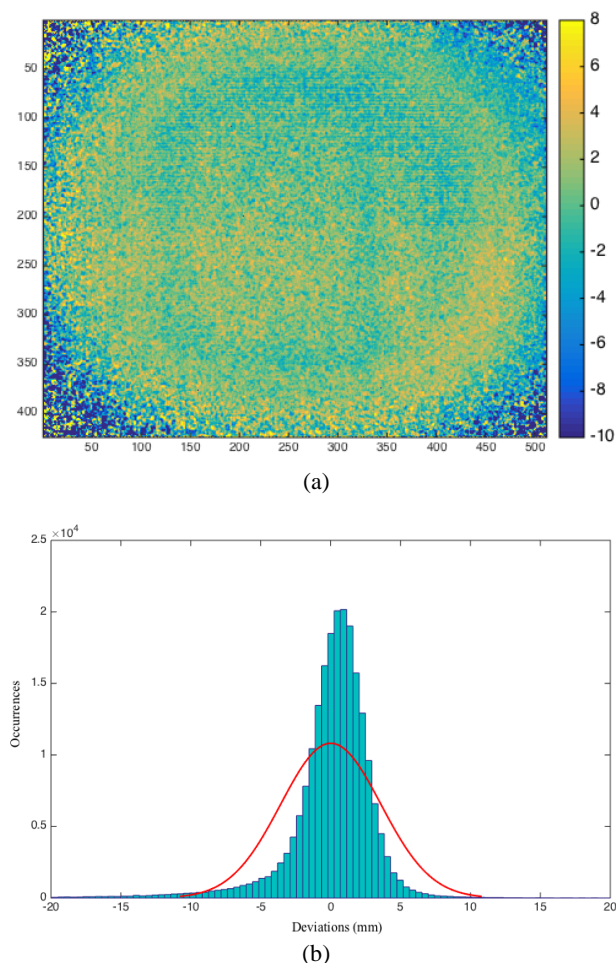


Figure 12. Residual distances between points and a fitted plane at the range of 1,25 m: (a) map with colorbar in millimeters; (b) corresponding histogram

Because of this loss of precision and reliability of the measurements at the boundaries, a first approach to reduce depth errors of entire sensor consists on considering only a restricted central array. Tests were realized with a reduced depthmap size of approximately 360x300 pixels from the 512x424 initial frame. Obviously, the area was chosen very central to remove border effects. The same phenomenon can always be detected at the boundaries of the new considered area but with widely reduced residuals. They reach a few millimeters at the range of 1,25 m. In addition, the standard deviations calculated for the fitted planes on the shorter areas were reduced from about 2 to 5 mm.

Nevertheless, a correction approach for the entire sensor area as proposed in (Pagliari et al., 2014) can be adapted to the device investigated here. It consists on delivering signed corrections interpolated from the residuals matrices computed at different ranges. For each pixel of the array, given the distance value it contains, a correction value will be interpolated considering the nearest ranges for which residuals were computed. The obtained correction matrix then just needs to be added to the initial depthmap in order to apply the per-pixel correction.

A two-way conclusion arises from these observations. On the one hand, using the inspected sensor for high accuracy metrology tasks justifies the need of a per pixel correction

managed over the whole sensor. Nevertheless, this correction step is time-consuming due to the local treatment of each pixel. Thus, on the other hand, people who foresee to use the Kinect for Windows v2 for lower-accuracy object reconstruction should locate their object on a central and restricted area of the sensor.

5. DISCUSSION

5.1 Main Advantages of Kinect v2

Thanks to the change of technology carried out by Microsoft for the release of its recent Kinect for Windows v2 sensor, the depth measurements achieved are more accurate compared to the first Kinect device. In addition, the computed point clouds present a better resolution because a distance measurement occurs for each pixel of the captured depthmaps. This allows a more accurate detection of small objects for example. The quality of color images has been also increased. These are the two main technical advantages of Kinect v2 compared to Kinect v1.

Tests about the influence of measurements repetition have shown that the averaging procedure has not necessarily a great influence on the final accuracy of the measurements. Indeed, the noise inherent to the sensor is already reduced in comparison to the first Kinect device, and is essentially present at objects boundaries where artifacts are hardly avoidable. Moreover, it has been shown that outdoor acquisitions can be envisaged. Kinect v2 device seems to be much less sensitive to daylight since point clouds were obtained even on a very sunny day. However, some filters need to be applied to overcome the increased noise effects and to reach exploitable results.

Another advantage is the simultaneous release of Kinect sensor and of its official SDK, which allows development of solutions for many tasks such as algorithms for 3D reconstruction.

Nonetheless, a few material drawbacks related to the use of the Kinect v2 sensor itself can be mentioned, such as the necessity of working under a Windows 8 or 8.1 station equipped with USB3 port. The required additional power input makes it also less mobile for SLAM applications for example.

5.2 Future Works

To complete our current work, it would be interesting to check the potential of the Kinect v2 device for small object modelling purposes. In this context, point clouds of a limestone fragment (approximately 25 x 20 cm) have already been captured. The acquisitions were made under different points of view all around the stone. Visual results look very satisfying, as it can be seen on Figure 13. The multiple point clouds (eight different positions) of the object now need to be registered in order to obtain a complete 3D point cloud. A first approach will consist on a manual segmentation of each individual point cloud in order to remove the artefacts (or flying pixels) that are clearly visible at the edges and boundaries of the object. Once the final mesh of this object will be produced, an accuracy assessment will be carried out thanks to comparisons with reference meshes acquired by other imaging techniques.

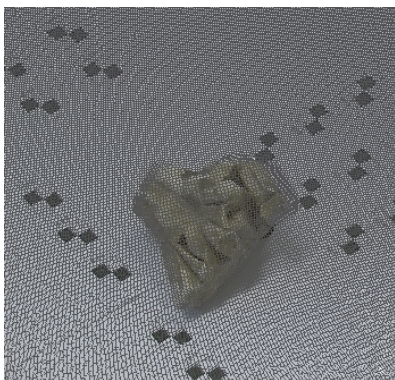


Figure 13. Point cloud of a limestone fragment obtained with Kinect v2

In a second time, tests will be performed using the Kinect Fusion tool mentioned before. It enables to directly acquire a meshed representation of the investigated object. This approach has the advantage to use much less point clouds as the manual approach and is a faster modelling method. However the model accuracy achieved through this method must be assessed. For sure, it depends on the way the Kinect is moved around the object.

6. CONCLUSION

The aim of this paper was to investigate through a set of tests the capacity of the Kinect v2 sensor, a low cost sensor, and to study if it could be an alternative to close range laser scanners for 3D measurements.

Several tests highlighted errors related to environment and to the properties of the captured object but also errors related to the system itself. Moreover, geometric as well as depth calibrations were carried out. Based on the obtained results, the physical limitations of the device could be assessed.

Considering depth measurement precision and outdoor efficiency, the achieved results look promising. Accordingly, the arrival of the second Kinect version can be seen as a real progress for computer vision tasks dealing with range imaging cameras.

In the future, the potential of the Kinect v2 device for small object modelling purposes will be assessed in a qualitative and a quantitative way.

REFERENCES

Bouguet, J.Y., 2013. Camera Calibration Toolbox for Matlab. http://www.vision.caltech.edu/bouguetj/calib_doc (2 Dec. 2013).

Dal Mutto, C., Zanuttigh, P., & Cortelazzo, G.M., 2013. Time-of-Flight Cameras and Microsoft Kinect™. Springer, Berlin, pp. 108.

Grzegorzek, M., Theobalt, C., Koch, R., & Kolb, A., 2013. Time-of-Flight and Depth Imaging - Sensors, Algorithms and Applications. Springer, Berlin, pp. 319.

Hansard, M., Horaud, R., Amat, M., & Evangelidis, G., 2014. Automatic detection of calibration grids in time-of-flight images. *Computer Vision and Image Understanding*, 121, 108-118.

Hansard, M., Lee, S., Choi, O., & Horaud, R., 2012. Time of Flight Cameras: Principles, Methods, and Applications. Springer, Berlin, pp. 95.

Kahlmann, T., Remondino, F., & Ingensand, H., 2006. Calibration for increased accuracy of the range imaging camera SwissRanger™. *Image Engineering and Vision Metrology (IEVM)*, 36(3), 136-141.

Kolb, A., Barth, E., Koch, R., & Larsen, R., 2009. Time-of-Flight Sensors in Computer Graphics. *Computer Graphics Forum*, 29(1), 141-159. doi:10.1111/j.1467-8659.2009.01583.x

Kolb, A., & Koch, R., 2009. Dynamic 3D Imaging. Springer, Berlin, pp. 117.

Lange, R., & Seitz, P., 2001. Solid-state time-of-flight range camera. *Quantum Electronics, IEEE Journal of*, 37(3), 390-397.

Lindner, M., & Kolb, A., 2006. Lateral and depth calibration of PMD-distance sensors. In: *Advances in Visual Computing*. Springer Berlin Heidelberg, pp. 524-533.

Lindner, M., Schiller, I., Kolb, A., & Koch, R., 2010. Time-of-flight sensor calibration for accurate range sensing. *Computer Vision and Image Understanding*, 114(12), 1318-1328.

Luhmann, T., Stuart, R., Kyle, S., Boehm, J., 2013. Close-range Photogrammetry and 3D Imaging. De Gruyter, Berlin, pp. 702.

May, S., Surmann, H., Perv, K., & Augustin, D.-S., 2006. 3D time-of-flight cameras for mobile robotics. *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 790-795. doi:10.1109/IROS.2006.281670

Mittet, M.-A., Grussenmeyer, P., Landes, T., Yang, Y. and Bernard, N., 2013. Mobile outdoor relative localization using calibrated RGB-D cameras. In: *8th International Symposium on Mobile Mapping Technology*, pp. 2-7.

Pagliari, D., Menna, F., Roncella, R., Remondino, F., & Pinto, L., 2014. Kinect Fusion improvement using depth camera calibration. In: *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 40(5), 479-485.

Remondino, F., & Stoppa, D., 2013. TOF Range-Imaging Cameras. Springer, 240 pages.

Reshetyuk, Y., 2009. Terrestrial laser scanning: error sources, self-calibration and direct georeferencing. VDM publishing house ltd, 161 pages.