

SPRAWDZENIE MOŻLIWOŚCI WYKORZYSTANIA KONTROLERA KINECT JAKO SKANERA 3D DO REJESTRACJI KOLOROWYCH CHMUR PUNKTÓW

CHECKING THE POSSIBILITIES OF KINECT CONTROLLER AS A SCANNER FOR REGISTRATION OF COLOR 3D POINT CLOUDS

Piotr Podlasiak

Politechnika Warszawska, Wydział GK, Zakład Fotogrametrii, Teledetekcji i SIP

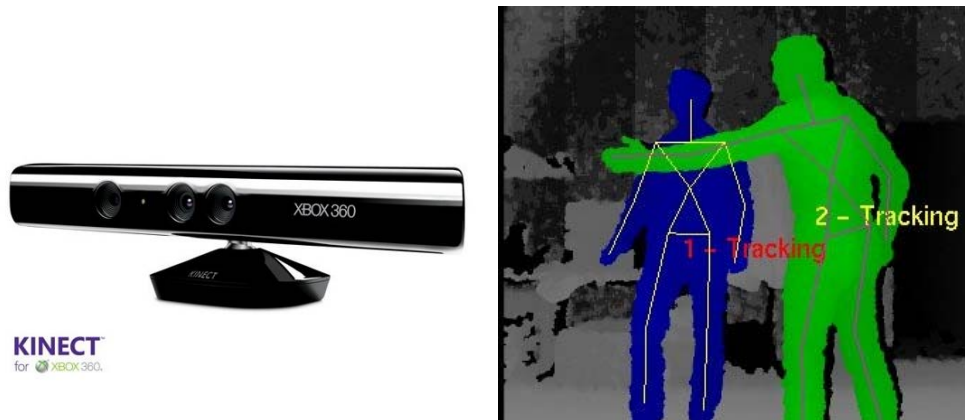
SŁOWA KLUCZOWE: skaning, światło strukturalne, Kinect, chmura punktów, DTM

STRESZCZENIE: Kontroler Kinect jest czujnikiem ruchu będącym urządzeniem wejściowym do konsoli gier XBOX (Microsoft®) pozwalającym na sterowane za pomocą gestów i poleceń słownych. Konstrukcja urządzenia zawiera dwie kamery (RGB i IR) oraz laserowy projektor podczerwieni. Procesor zawarty w urządzeniu na podstawie obrazu z kamery IR wyznacza „mapę głębokości” (640 x 480) w zakresie od 0.8 m do 5 m w czasie rzeczywistym (30 Hz). Niska cena urządzenia w porównaniu do jego możliwości sugeruje zastosowanie go jako odpowiednik skanerów laserowych 3D bliskiego zasięgu. W pracy przedstawiono szereg zagadnień związanych z pobieraniem informacji z urządzenia do komputera PC, generowaniem chmury punktów i jej kolorowaniem tzn. łączeniem obrazu z kamer RGB i IR. W czasie praktycznych prób okazało się, że format informacji jest nietypowy a dane dostarczane są ze zmienną dokładnością zależną od odległości od urządzenia. Konieczne stało się opracowanie własnych metod kalibracji sensora „głębokości” uwzględniających specyfikę działania urządzenia i cechy otrzymywanej chmury punktów. Istotnym problemem w kalibracji był brak możliwości wykonywania jej metodą klasyczną na „punktach” terenowych, które nie ulegały odwzorowaniu na mapie głębokości. Szereg problemów należało rozwiązać przy łączeniu obrazu „głębokościowego” z widzialnym ze względu na różne rozmieszczenie, rozdzielczość i pole widzenia obu kamer. W drugiej części pracy przedstawione zostanie metoda kalibracji i jej wyniki dla konkretnego egzemplarza czujnika Kinect oraz uzyskane dokładności. W celu sprawdzenia „jakości” uzyskanej chmury punktów przeprowadzono także porównanie wyników skaningu uzyskanych innymi bardziej profesjonalnymi urządzeniami. Uzyskane wyniki pozwalają na oszacowanie dokładności pomiaru rzędu kilku mm przy jednocześnie dużej nieciągłości pomiarów – mapa głębokości zawiera tylko około tysiąca różnych wartości.

1. INFORMACJE OGÓLNE O KONTROLERZE KINECT

Kontroler Kinect został przedstawiony po raz pierwszy w czerwcu 2009 na targach E3 a wprowadzony do sprzedaży pod koniec 2010. Został on wpisany do księgi rekordów Guinnessa jako "najszybciej sprzedające się urządzenie elektroniki konsumenckiej", po tym, jak w przeciągu pierwszych 60 dni został sprzedany w ilości przekraczającej 8 milionów egzemplarzy. Jest to kontroler zaprojektowany do współpracy z konsolą gier

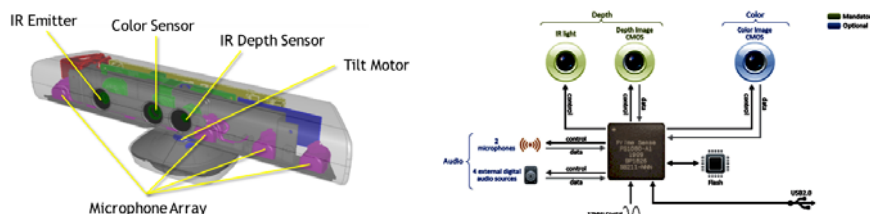
XBOX 360 pozwalający na sterowanie za pomocą gestów i poleceń słownych. KINECT ma możliwość jednoczesnego śledzenia dowolnej liczby postaci, w tym 2 aktywnych graczy, będących w stanie zmieścić się w polu widzenia kamery. Urządzenie, śledząc postacie i dokonując analizy ich ruchów, "dzieli" człowieka na 48 części, monitorowanych w czasie rzeczywistym. Ponieważ urządzenie podłączone jest do konsoli za pomocą standardowego złącza USB pojawiła się koncepcja wykorzystania go do współpracy z komputerami PC. Hakerzy (w pozytywnym znaczeniu tego słowa) zbadali protokoły transmisji danych i opublikowali wyniki analiz w Internecie co pozwoliło na stworzenie kilku bibliotek pozwalających na odczyt danych z urządzenia. Wielkie zainteresowanie zmusiło także Microsoft w czerwcu 2011 do opublikowania SDK (Software Development Kit) pozwalającego na (ograniczone) programowanie. W 2012 pojawiła się nowa wersja oprogramowania a także nowa odmiana sprzętowa Kinect for Windows (do 9 grudnia 2012 nie była jeszcze dostępna w Polsce).



Rys. 1. Widok zewnętrzny kontrolera i przykład śledzenia postaci

1.1. Budowa i dane techniczne

Koncepcja kontrolera Kinect została oparta na patentach firmy PrimeSense (PrimeSense). Z danych przedstawionych na stronach internetowych producenta (Microsoft) i innych (głównie Openkinect i ROS) wynika, że wewnątrz obudowy znajdują się:



Rys. 2. Wnętrze kontrolera Kinect według Microsoft (po lewej) i według PrimeSense (po prawej)

- *IR Emitter* – projektor laserowy (długość fali 830 nm, około 60 mW mocy), wyposażony w układ stabilizujący temperaturę (ogniwo Peltiera) ,
- *Color Sensor* - cyfrowy przetwornik obrazu - kamery z wyjściem o rozdzielczości wg. kart katalogowych 1280 x 1024 pikseli, 30 obrazów na sekundę, zapis 10 bitów z matrycą Bayera,
- *IR Depth Sensor* – monochromatyczny cyfrowy przetwornik obrazu (analogiczny do kolorowego) wyposażony w filtr przepuszczający światło w zakresie bliskiej podczerwieni i blokujący obszar widzialny.
- *Tilt Motor* – silnik sterujący pochYLENIEM głowicy (zakres +/- 27 stopni)
- *Microphone Array* – zestaw mikrofonów

oraz nie pokazane na rysunku:

- akcelerometr trzyosiowy do wyznaczenia kierunku pionu,
- układy sterujące, procesor z pamięcią i układy transmisji USB, wentylator itp.

Parametry czujnika głębokości: pole widzenia w poziomie 57 stopni, w pionie 47 stopni, zasięg działania od 0.8 m do 4.5 m.

Całkowity pobór mocy wynosi około 800 mA co wymaga dodatkowego zasilacza. Łącze USB wykorzystuje 70-80% przepustowości koncentratora (*Host Controller*) USB 2.0 co powoduje, że do tego samego *Hub'a* (czyli kilku gniazd) nie powinno się podłączać innego urządzenia wymagającego szybkiej transmisji jak pendrive, dysk lub kamera internetowa.

1.2. Oprogramowanie

Początkowo Microsoft nie zgadzał się na wykorzystanie Kinecta w połączeniu z innymi urządzeniami niż XBOX 360 (zastrzeżenie takie znajdowało się w instrukcji zakupionego przeze mnie urządzenia), potem jednak udostępniał SDK w kolejnych wersjach. Oprogramowanie ma jednak postać czarnej skrzynki, dostarczającej informacji o przestrzeni 3D (początkowo o ograniczonej rozdzielczości do 320 x 240 „pikseli”) bez szczegółów dotyczących uzyskiwanych dokładności – zresztą absolutnie zbędnej w przypadku gier. Dodatkowo biblioteki wymagały systemu Windows 7 i kompilatora Visual C# (C sharp) co skłoniła autora do poszukiwania innych rozwiązań. Po zapoznaniu się ze sporymi zasobami Internetu całe oprogramowanie wykorzystane do uzyskania prezentowanych dalej wyników zostało napisane w oparciu o informacje ze stron poświęconych temu zagadnieniu oraz darmowych bibliotek libusb_win32 (do komunikacji USB) i OpenCV (do przetwarzania obrazu). Stworzone przez autora oprogramowanie pozwala na odczyt z częstotliwością 30 Hz:

A.obrazu RGB 640 x 480 x 24 bitów

B.obrazu IR 640 x 488 x 8 bitów

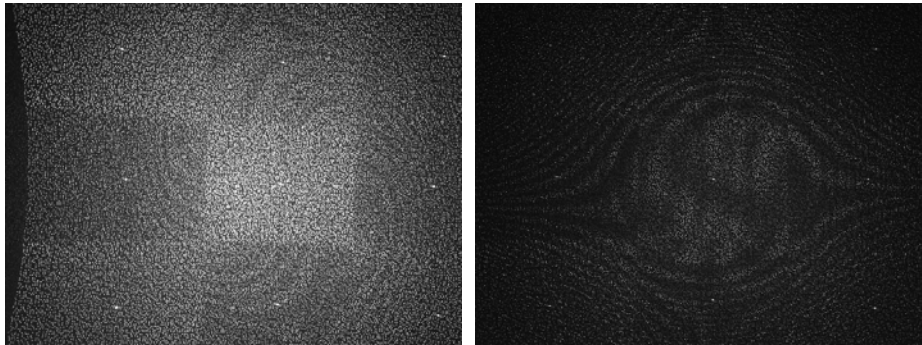
C.mapy głębokości 640 x 480 x 16 bitów (11 bitów informacji)

a także odczyt danych z akcelerometru i sterowanie pochYLENIEM głowicy.

Kolejny moduł oprogramowania pozwala na kalibrację na podstawie zarejestrowanych obrazów i przedstawienie wyników w postaci chmur punktów 3D.

2. ZASADA DZIAŁANIA

Przedstawione dalej zasady działania nie są oparte na informacjach producenta lecz na analizach zaczerpniętych z Internetu i doświadczeniu własnym autora. Obserwacja działania kontrolera pokazała, iż informacje o głębokości uzyskiwana jest za pomocą wykorzystania światła strukturalnego w zakresie niewidocznym dla człowieka. Projektor podczerwieni wyświetla zdefiniowany w urządzeniu wzór „kropek”, dokładniej powtórzony 9 krotnie wzorzec o rozmiarze 211 x 165 pikseli (razem 633 x 495) z których około 11 % jest rozjaśnionych (Kinect Pattern).



Rys. 3. Przykłady wzorów „kropek” tworzonych przez projektor Kinect i rejestrowanych przez kamerę IR (widoczne silne zniekształcenia wynikające z dystorsji obiektywu)

Najprawdopodobniej specjalizowany procesor PrimeSense 1080 koreluje obraz z kamery IR z zapisanym wzorcem (z pewnych cech przesyłanych można podejrzewać, że rozmiar okna jest 9x9 przy rozdzielczości obrazu 1280x1024) i wyznacza paralaksę podłużną (*disparity*) z subpikselową (1/8) dokładnością. Ciekawostką jest, iż obrazy dostarczane przez kontroler nie odpowiadają rozdzielczościom przetworników i dla częstotliwości przetwarzania 30 obrazów na sekundę wynoszą:

- obraz kolorowy 640 x 480 pikseli x 8 bitów (konieczne jest odtworzenie kolorów z macierzy Bayera)
- obraz w podczerwieni: 640 x 488 pikseli x 8 bitów (można otrzymywać albo RGB albo IR, nie ma jednoczesnej rejestracji)
- mapa głębokości: 640 x 480 pikseli x 11 bitów, ale ostatnie 8 pikseli w wierszu nie zawiera informacji więc rozdzielczość wynosi praktycznie 632 x 480.

3. CHMURA PUNKTÓW

3.1. Generowanie chmury punktów

Kontroler dostarcza poprzez łącze USB 30 razy na sekundę dwa bloki danych: obraz RGB i skompresowaną mapę głębokości. W celu przekształcenia tych informacji w chmurę punktów konieczne jest wykonanie działań przedstawionych w tabeli 1.

Tabela 1. Sposób wyznaczenia chmury punktów (X,Y,Z oznaczają współrzędne terenowe, u,v – współrzędne pikselowe obrazu, indeksy IR – kamerę podczerwoną, RGB kamerę kolorową)

Lp.	Etap	Niezbędne informacje
1	odczyt paralaksy z mapy głębokości $d = d[u_{IR}, v_{IR}]$	mapa głębokości
2	wyznaczenie rzeczywistej „głębokości” Z_{IR}	dane kalibracji odległości
3	wyznaczenie współrzędnych X_{IR}, Y_{IR} punktu	Z_{IR} + dane kalibracji kamery IR
4	przeliczenie X_{IR}, Y_{IR}, Z_{IR} do układu kamery RGB $= X_{RGB}, Y_{RGB}, Z_{RGB}$	orientacja wzajemna kamer IR <-> RGB
5	wyznaczenie współrzędnych $[u_{RGB}, v_{RGB}]$ obrazu punktu w kamerze RGB,	dane kalibracji kamery RGB
6	nadanie przetwarzanemu punktowi koloru odczytanego z obrazu RGB	obraz RGB

Sposób wyznaczenia potrzebnych informacji zostanie pokazany w rozdziale 4.

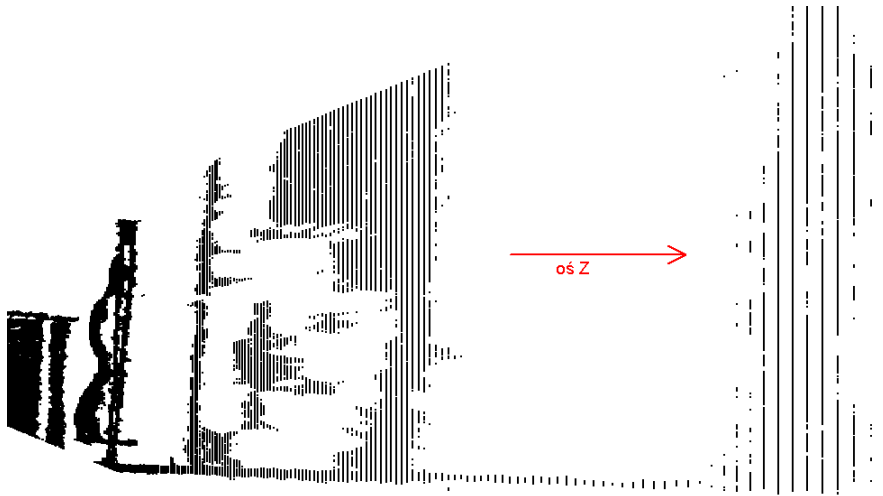
3.2. Specyfika uzyskanej chmury

Wszystkie przedstawiane dane są przekazywane w postaci cyfrowej a więc o ograniczonej precyzji. Ograniczenie to jest szczególnie istotne w przypadku kierunku „głębokości”, gdzie paralaksa może przyjmować teoretycznie 11 bitów czyli 2047 wartości (wartość maksymalna = 2047 jest wykorzystywana jako sygnalizacja braku danych), w praktyce nieco powyżej tysiąca różnych wartości. Odbija się to na uzyskiwanej rozdzielczości, która silnie pogarsza się ze wzrostem odległości od kontrolera (tabela 2).

Tabela 2. Zależność rozdzielczości chmury punktów od odległości

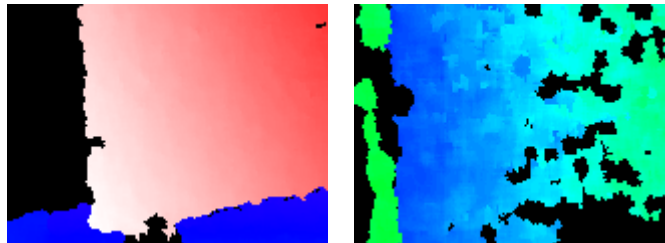
Odległość Z [m]	Paralaksa [piksele]	Rozdzielczość Z [mm]	Rozmiar X,Y [mm]	Rozmiar RGB [mm]
0.50	396	0.7	0.8	1.0
1.00	744	2.9	1.7	1.9
1.50	860	6.4	2.5	2.9
2.00	918	11.4	3.4	3.8
2.50	952	17.8	4.2	4.8
3.00	976	25.6	5.1	5.7
3.50	992	34.9	5.9	6.7
4.00	1005	45.5	6.8	7.6
4.50	1014	57.4	7.6	8.6
5.00	1022	70.8	8.5	9.5

Zjawisko to jest widoczne podczas obserwacji uzyskanych chmur np. w płaszczyźnie XZ lub YZ, jak pokazano to na rysunku 4.



Rys. 4. Skokowe zmiany współrzędnej Z chmury punktów

Kolejna niedogodność wynika z faktu iż korelacja jest wykonywana tylko dla „rozświetlonych” kropek stanowiących zaledwie 11 % powierzchni (rys. 3) i tylko dla nich wartość paralaksy (a więc i Z) jest określona poprawnie, pozostałe wartości są wyznaczone przez nieznaną algorytm „zaszyty” wewnątrz urządzenia. Powoduje to błędy w pobliżu krawędzi obiektów a w niesprzyjających warunkach oświetleniowych powoduje „dziury” w mapie głębokości (rys. 5).



Rys. 5. Przykłady błędów działania algorytmu tworzącego mapę głębokości

Pozytywną cechą algorytmu jest fakt iż punkty w chmurze są uporządkowane: od lewej do prawej, od góry do dołu co ułatwia triangulację (tworzenie siatek *mesh*).

4. KALIBRACJA I UZYSKANE DOKŁADNOŚCI

Przedstawione wcześniej obrazy i chęć zbadania dokładności wymagały wykonania kalibracji obu stosowanych kamer. Konieczne było także wyznaczenie funkcji opisującej zależność zwracanej przez kontroler głębokości (*depth*) od rzeczywistej odległości obiektu od kontrolera. Sporym problemem okazało się także wyznaczenie wzajemnego położenia kamer niezbędne do prawidłowego kolorowania chmury punktów.

4.1. Kalibracja obrazów

Kalibracja kamery została wykonana z wykorzystaniem biblioteki OpenCV i jej procedur kalibracji opartych na rejestracji kilkunastu obrazów czarno-białej szachownicy (oprogramowanie wykrywa narożniki z dokładnością podpikselową). Podobną metodę wykorzystano dla kamery IR, choć okazało się że rzutowane kropki zakłócają identyfikację narożników, rozwiązaniem okazało się przesłonięcie projektora podczerwieni i zastosowanie silnego oświetlenia widzialnego. OpenCV opisuje dystorsję za pomocą modelu parametrycznego (OpenCVb). Kalibracja z wyborem różnej liczby współczynników dystorsji (na podstawie 20 obrazów szachownicy) pozwoliła na wysnucie ciekawych wniosków dotyczących współczynników dystorsji. Na rysunku 6 przedstawiono przykładowe obrazy wykorzystywane do kalibracji (po lewej stronie kamera RGB, po prawej IR).

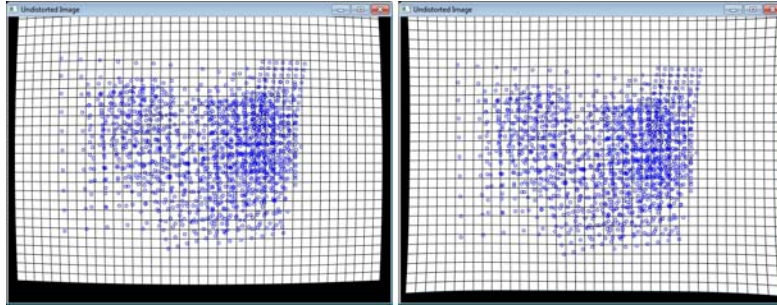


Rys. 6. Przykładowe obrazy użyte do kalibracji

Tabela 3 przedstawia uzyskane wyniki liczbowe. Warto zwrócić uwagę że wybór modelu kalibracji w niewielkim stopniu wpływa na błąd odtworzenia położenia narożników (reprojection error), natomiast wartości liczbowe ulegają dużym zmianom (f_x, f_y stałe kamery, c_x, c_y położenie punktu głównego (OpenCVb)). Na rysunku 7 pokazano także wpływ tych parametrów na kształt obrazu po korekcji, obraz po lewej stronie to stan przy uwzględnieniu tylko pierwszego współczynnika k_1 , po prawej wszystkich. Niebieskie kółka pokazują położenie wszystkich narożników, jak widać na wyniki mógł mieć wpływ brak punktów pomiarowych w pobliżu brzegów zdjęcia.

Tabela 3. Wyniki kalibracji kamer

Parametr	jednostki/ parametry	Kamera RGB		Kamera IR	
		tyko k_1	wszystkie	tyko k_1	wszystkie
(<i>reprojection error</i>)	piksele	0.251082	0.201738	0.381925	0.329281
f_x	piksele	536.046	534.548	589.107	592.207
f_y	piksele	533.846	532.039	590.530	593.905
c_x	piksele	1.794	-3.322	5.605	8.064
c_y	piksele	26.814	17.059	4.444	-13.636
k_1		0.120019	0.245238	-0.255264	-0.145527
k_2			-0.673976		0.897962
k_3			0.420228		1.672420
p_1			-0.00809668		-0.0111527
p_2			-0.00784751		0.00131223



Rys. 7. Wpływ wyboru liczby parametrów dystorsji na kształt poprawionego (*undistort*) obrazu

4.2. Kalibracja funkcji odległości

Hipoteza iż dane zawarte w mapie głębokości odpowiadają paralaksie podłużnej wymagała potwierdzenia. Dla przypadku normalnego pary zdjęć (oznaczenia stosowane w fotogrametrii naziemnej):

$$z = \frac{Bc_k}{p_x} \quad (1)$$

Zakres wartości paralaksy jest ograniczony do 11 bitów (0-2047) co sugeruje konieczność przesunięcia zakresu wartości o pewną stałą. Wprowadzając inne oznaczenia: d_p – paralaksa (*disparity*), d_0 – stałe przesunięcie, *const* – stała otrzymamy postać wzoru:

$$z = \frac{const}{d_p - d_0} \quad (2)$$

Po wykonaniu rejestracji prostopadłościaków ustawionych w różnych odległościach uzyskano następujące wartości stałych (w celu uzyskania wyniku w m):

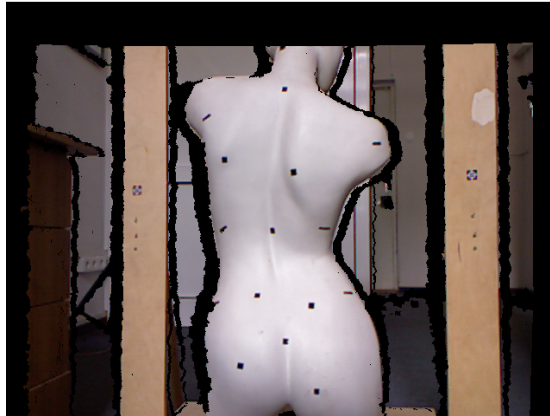
$$d_0 = 1084, \quad const = 348.616,$$

Warto zauważyć że odległość między kamerą a projektorem IR wynosi 0.075 m, $c_k = 589$ pikseli co potwierdza hipotezę o subpikselowej (1/8) rozdzielczości.

$$const = 348.616 \approx 0.075 * 589 * 8 = 353.3 = 8 * B * c_k$$

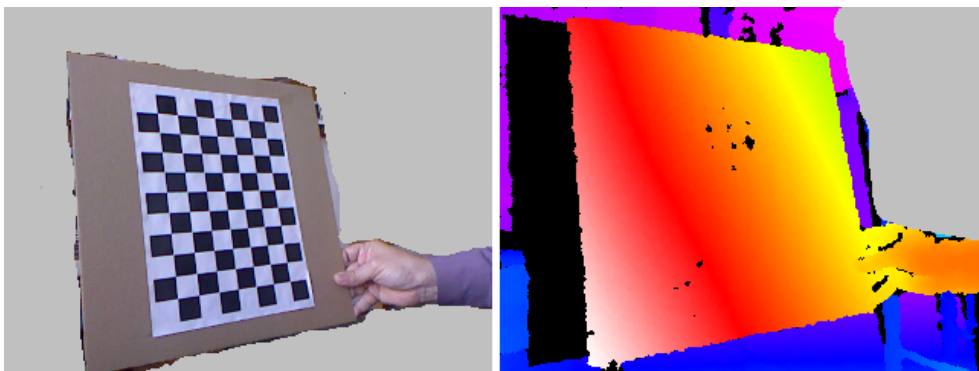
4.3. Wyznaczenie wzajemnego położenia kamer

W celu prawidłowego „pokolorowania” konieczne jest określenie wzajemnego położenia kamer IR i RGB. Na rysunku 8 pokazano, że pola widzenia nie pokrywają się, kamera głębokości widzi mniejszy obszar niż RGB.



Rys 8. Na obrazie RGB zaczerniono piksele które nie posiadają informacji o odległości (obraz powstał już po wykonaniu kalibracji)

Dla wyznaczenia wzajemnego usytuowania kamer nie było możliwe wykorzystanie danych kalibracyjnych kamery IR ponieważ rozmiar obrazu tej kamery 640x488 jest inny niż mapy głębokości: 632 x 480. Proste przesunięcie obrazu o -4 piksele w każdym z kierunków nie dało poprawnych wyników. Zastosowanie klasycznej metody orientacji wzajemnej nie było możliwe z uwagi na fakt, iż na obrazie mapy głębokości nie da się zidentyfikować punktów wspólnych z punktami obrazu widzialnego (rys. 9). Dopiero wyznaczenie równań płaszczyzn na których leżały szachownice pozwoliło na wyznaczenie przemieszczeń kątowych a „wirtualne” punkty przecięcia „trójek” płaszczyzn posłużyły do wyznaczenia parametrów liniowych.



Rys. 9. Obraz RGB i mapa głębokości dla tego samego pomiaru

4.4. Dane kalibracyjne zawarte w urządzeniu

Po samodzielnym wyznaczeniu opisanych parametrów udało się dotrzeć do informacji o zakodowanych w pamięci urządzenia (indywidualnych dla egzemplarza) parametrach „kalibracji” obrazów i głębokości. Po eksperymentach okazało się jednak, że ze względu na

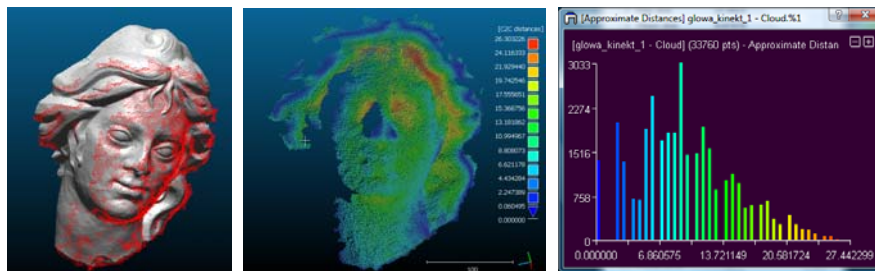
arytmetykę stałoprzecinkową uzyskane korekcja jest gorsza (czynnik 2) od wyznaczonej eksperymentalnie.

5. PRZYKŁADOWE POMIARY

Dla oceny rzeczywistej dokładności przeprowadzono eksperymenty na obiektach dla których istniały inne dane ze skaningu bardziej profesjonalnym i sprawdzonym sprzętem. Do obróbki i porównywania wykorzystano oprogramowanie open source: CloudCompare (CloudCompare).

5.1. Głowa

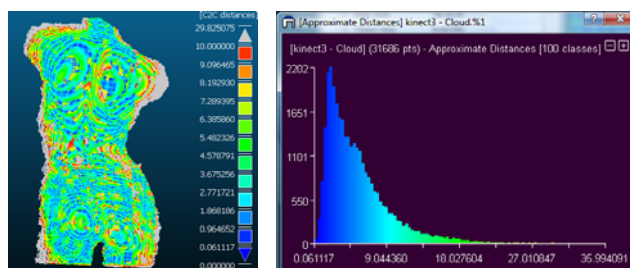
Badanym obiektem była fragment kamiennej rzeźby, dla której istniały dane ze skaningu skanerem ze światłem strukturalnym firmy SMARTTECH zawierające ok. 1 mln punktów pozyskanych z 7 skanów. Według danych producenta skanera rozdzielczość chmury punktów wynosi od jednego do kilkuset punktów na mm^2 – informacji o dokładności nie udało się uzyskać. Do porównania wykorzystano rejestrację Kinect z 33 tys. punktów. Rysunek 10 przedstawia oceny rzeczywistej dokładności przeprowadzono eksperymenty na obiektach dla których istniały inne dane ze skaningu bardziej profesjonalnym i sprawdzonym sprzętem. Średni błąd wyniósł 5 mm, widoczna na wykresie skokowa zmiana odchyłek.



Rys. 10. Wpasowanie punktów Kinect (czerwone punkty) i rozkład odchyłek na powierzchni modelu – obrazy uzyskane za pomocą programu CloudCompare

5.2. Manekin

W drugim eksperymencie wykorzystano manekin używany do testowania oprogramowania medycznego. Porównywano skan Kinect (około 32 tys. pkt.) z danymi uzyskanymi metodą mory (21 tys. pkt.) – rozdzielczość w płaszczyźnie XY 2-3 mm, w kierunku osi Z teoretycznie 0.1 mm, praktycznie 0.7 mm (Podlasiak *et al.*, 2003). Średnia różnica głębokości wynosiła 4 mm (na rysunku 11 widać że największe różnice pojawiają się na określonych warstwicach co może być efektem niedokładności każdej z metod pomiaru).



Rys. 11. Rozkład odchylek na powierzchni modelu – obrazy uzyskane za pomocą programu CloudCompare

6. WNIOSKI

Kontroler Kinect jest przeznaczony do zastosowania w dynamicznych grach komputerowych co powoduje nastawienie na szybkość z mniejszym wymaganiem dokładnościowym. Osiągnięte wyniki nie pozwalają na razie na wykorzystanie w zastosowaniach dokumentacyjnych ale w zupełności wystarczy do zastosowań medycznych gdzie wystarczyłaby dokładność 1 cm (informacja uzyskana na podstawie kontaktów z lekarzami ortopedami). Autor artykułu był jednym z twórców Systemu Analizy Wad Postawy (patent PL 170284, Podlasiak *et al.*, 2002) wykorzystującego kosztowne urządzenie projekcyjne, które mogłoby być zastąpione przez kontroler Kinect dający możliwość badania w ruchu, w czasie rzeczywistym. Wyniki takich eksperymentów zostaną zapewne przedstawione w kolejnych publikacjach.

Pewne nadzieje daje także nowa wersja *Kinect for Windows* która ma działać w mniejszej odległości a więc z większą rozdzielczością. Konieczne są także dalsze prace nad zbadaniem zaszytych w urządzeniu metod interpolacji.

7. LITERATURA

- CloudCompare, 2012 <http://www.danielgm.net/cc/>
Kinect Pattern, 2012 <http://azttm.wordpress.com/2011/04/03/kinect-pattern-uncovered/>
Libusb, 2012 <http://sourceforge.net/apps/trac/libusb-win32/wiki/>
Microsoft, 2012 <http://www.microsoft.com/en-us/kinectforwindows/>
OpenCVa, 2012: <http://opencv.willowgarage.com/wiki/>
OpenCVb, 2012 http://docs.opencv.org/doc/tutorials/calib3d/camera_calibration/camera_calibration.html
OpenKinect, 2012 [http:// openkinect.org](http://openkinect.org)
PrimeSense, 2012 <http://www.primesense.com>
ROS (RobotOperatingSystem), 2012: http://www.ros.org/wiki/kinect_node
Podlasiak P. Zawieska D., 2002: System automatycznej analizy prążków mory projekcyjnej, *Archiwum Fotogrametrii, Kartografii i Teledetekcji*, Vol. 12b, str. 376–387, Warszawa.

Podlasiak P. Zawieska D., 2003. Analiza czynników wpływających na dokładność pomiarów systemem mory projekcyjnej. *ArchiwumFotogrametrii, Kartografii i Teledetekcji*, Vol.13 B, s. 507-517.

CHECKING THE POSSIBILITIES OF KINECT CONTROLLER AS A SCANNER FOR REGISTRATION OF COLOR 3D POINT CLOUDS

KEYWORDS: scanning, structured light, Kinect, cloud points, DTM

Summary

The controller Kinect is a motion sensor which is an input device for the XBOX 360 game console (Microsoft ®) allows the controlled with gestures and verbal commands. Device has two cameras (RGB and IR) and IR laser projector. The processor included in the device based on IR camera image compute "depth map" (640 x 480) in the range from 0.8 m to 5 m in real time (30 Hz). The low price of the device compared to its features suggest its use as the equivalent of 3D laser scanners shipping. The paper presents a number of issues related to the collection of information from the device to a PC, point cloud generation and its coloring. During practical tests proved that the format is unusual information and data are provided with variable accuracy depending on the distance from the device. It became necessary to develop their own methods for sensor calibration "depth" specific to the operation and features of the obtained point cloud. A major problem in the calibration was not possible to exercise its classical way to "point" field, which may not yield a depth map projection. A number of problems had to be solved by combining depth image with visible due to the different distribution, resolution and field of view of both cameras. In the second part of the study will be presented calibration method and results for a specific copy of the Kinect sensor and the accuracy obtained. In order to check the "quality" of the resulting point cloud was carried out to compare the results of other more professional scanning methods. The obtained results allow us to estimate the accuracy of a few mm at the same time a large discontinuity measurements - depth map contains only about a thousand different values.

Dane autora:

Dr inż. Piotr Podlasiak
e-mail: ppodlasiak@wp.pl
telefon: 22 234 76 94