

# Skaner 3D z setką Raspberry Pi

*Nie jest trudno pokazać, że za pomocą miniaturowego Raspberry Pi można zrealizować wielkie rzeczy – projekty zarówno innowacyjne, jak i niedrogie, a ponadto łatwe do wykonania. Co w takim razie można zrobić, gdy ma się do dyspozycji np. 100 sztuk komputerka Raspberry Pi? Świetnym przykładem możliwości jest skaner 3D, opracowany przez Richarda Garsthagena z Holandii.*

Zacznijmy od uspokojenia Czytelników, że do budowy skanera 3D nie potrzeba aż setki komputerów. Właściwie to wystarczy jeden, ale kosztem znacznego zwiększenia wymagań odnośnie do zastosowanych elementów mechanicznych. Pojedyncza kamera musiałaby otaczać skanowany obiekt (lub musiałby się obracać na podstawie) np. po spirali, fotografując go z wielu punktów. Szybkość tego ruchu byłaby ograniczona przez precyzję mechaniki, czułość kamery oraz jasność oświetlenia, tak aby wykonywane fotografie nie były rozmazane (idealnie by było, gdyby kamera zatrzymywała się przed wykonaniem każdego zdjęcia). W efekcie konieczne byłoby użycie zaawansowanego systemu pozycjonowania kamery, a sam proces skanowania trwałby kilka-kilkanaście minut, w zależności od pożądanej dokładności skanu.

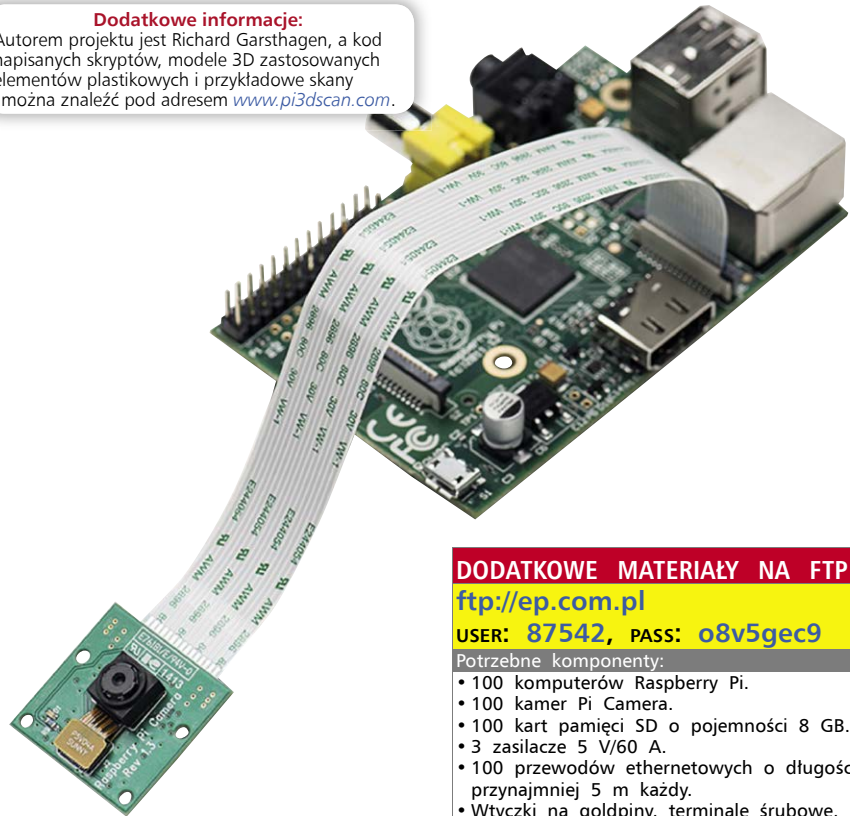
## Geneza pomysłu

Autorowi projektu zależało jednak na tym, by skan wykonywany był błyskawicznie, gdyż celem było wykonanie modeli 3D jego dzieci, które trudno było zmusić do nieruchomego przebywania w jednej pozycji. Stąd powstała konieczność zastosowania wielu kamer, które wykonywałyby fotografię jednocześnie. Pierwszym pomysłem było użycie tanich aparatów fotograficznych, dla których opracowany zostałby system automatycznego wyzwiania, ale wtedy poszczególne zdjęcia byłyby zapisane na oddzielnych kartach pamięci, co znacząco utrudniłoby przetwarzanie ich.

Na szczęście autor trafił na artykuł opisujący Raspberry Pi z modulem Pi Camera, które choć nie wydawały się najtańszym rozwiązaniem dla skanera, dawały dużą elastyczność pracy. Płytki można ze sobą połączyć przez Ethernet i szybko wyzwalać, a wykonane

### Dodatkowe informacje:

Autorem projektu jest Richard Garsthagen, a kod napisanych skryptów, modele 3D zastosowanych elementów plastikowych i przykładowe skany można znaleźć pod adresem [www.pi3dscan.com](http://www.pi3dscan.com).



Fotografia 1. Moduł Pi Camera

zdjęcia w łatwy sposób zgrać na jeden, centralny serwer. W efekcie Richard postanowił nabyć 40 sztuk Raspberry Pi i po jednym module Pi Camera do każdego z nich, 40 kart SD o pojemności 8 GB, zasilacz 5-woltowy o prądzie znamionowym 60 A, celem zasilania komputerów oraz zestaw taśm LEDowych wraz z mocnym, 12-woltowym zasilaczem, które miały oświetlać skanowany obiekt. Okazało się, że podłączenie już 12 kamer daje bardzo dobre rezultaty, a zwiększenie liczby modułów pozwala polepszyć precyzję. Koniec końców, autor przygotował zestaw z 115 komputerów z kamerami, z których każda wykonuje dwa 5-megapikselowe zdjęcia na jeden skan, dostarczając w sumie ponad miliard pikseli, służących do stworzenia wirtualnego modelu. Naturalnie wykonanie projektu wymagało rozwiązania całego szeregu problemów, począwszy od fizycznego połączenia ze sobą tak dużej liczby komputerów, a kończąc na przełożeniu wykonanych zdjęć na model 3D.

## Połączenia fizyczne

Autor podjął kilka różnych prób ustawienia sprzętu, przy czym zdecydowanie najwygodniejsze w użytkowaniu okazało się zamontowanie kamer na oddzielnych słupkach, ustawianych wokół skanowanego obiektu.

### DODATKOWE MATERIAŁY NA FTP:

<ftp://ep.com.pl>

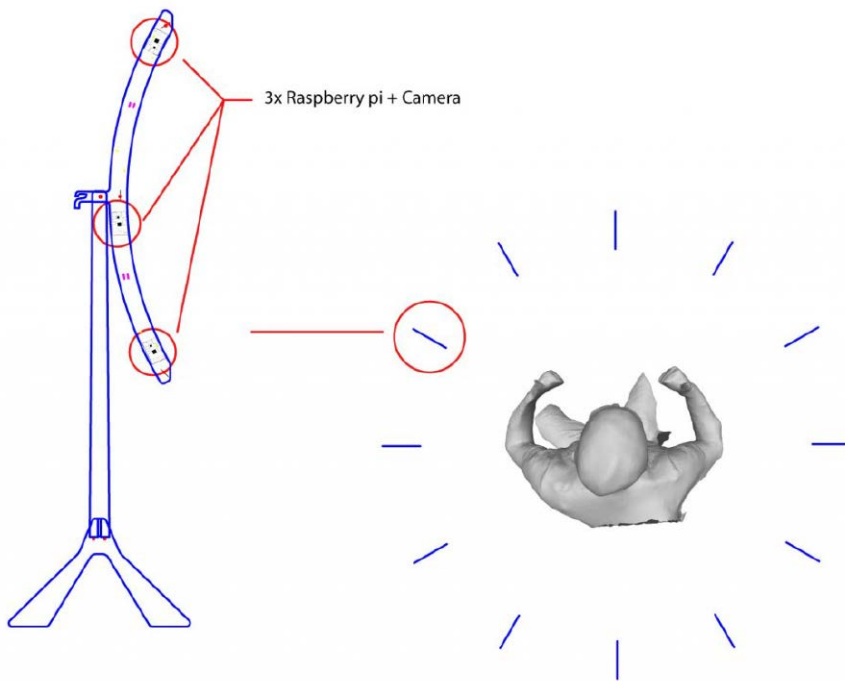
USER: 87542, PASS: o8v5g9c9

Potrzebne komponenty:

- 100 komputerów Raspberry Pi.
- 100 kamer Pi Camera.
- 100 kart pamięci SD o pojemności 8 GB.
- 3 zasilacze 5 V/60 A.
- 100 przewodów ethernetowych o długości przynajmniej 5 m każdy.
- Wtyczki na goldpiny, terminale śrubowe, dodatkowe przewody do przyłączenia zasilania itp.
- 3 szt. 48-portowych switchy ethernetowych.
- 8 szt. 5-metrowych taśm LED,
- Zasilacz 24 V/10 A.
- Słupki/stojaki.
- Śrubki.

Pierwsza wersja skanera miała słupki drewniane o łukowatym kształcie, na których były zamontowane po trzy kamery. W słupkach wywiercone zostały otwory do przykręcenia Raspberry Pi. Płytki modułu Pi Camera zostały natomiast zamontowane na niewielkich uchwytach, zaprojektowanych przez autora i wydrukowanych na drukarce 3D. Uchwyty te również zostały przykręcone do słupków. Dodatkowo, by uniezależnić jakość skanu od zewnętrznych warunków oświetleniowych, na każdym ze słupków zamontowano po 1-metrowej taśmie diodowej, z zainstalowanymi 60 białymi LEDami.

Niewątpliwie dużym wyzwaniem jest połączenie ze sobą oraz z zasilaniem kilkudziesięciu komputerów umieszczonych na przenośnych słupkach. Już przy 40 komputerach, poprowadzenie 40 przewodów sieciowych może być problematyczne, a dodatkowo pociągnięcie razem z nimi przewodów zasilających także nie będzie łatwe. Niestety, komputery Raspberry Pi nie obsługują technologii Power over Ethernet, która mogłaby znacznie ułatwić instalację. Autor postanowił jednak samodzielnie ją



Rysunek 2. Słupki użyty w jednej z pierwszych wersji projektu

zaimplementować (w podstawowym zakresie), wykorzystując nieużywane żyły w kablach sieciowych do przesyłania napięcia zasilającego.

Kable ethernetowe zostały podzielone na równe kawałki – na potrzeby początkowego ustawienia z 40 komputerami wystarczyły 5-metrowe odcinki. Równa długość wszystkich przewodów jest o tyle istotna, że nie są one idealnymi przewodnikami do zasilania, w związku z czym mogą powstawać na nich zbyt duże spadki napięcia, by móc zasilić Raspberry Pi z kamerą. Zastosowanie tej samej długości przewodów dla wszystkich komputerów sprawia, że spadek napięcia na złączu zasilania każdego z komputerów będzie identyczny. Można go zmierzyć i skorygować napięcie wspólnego zasilacza komputerów tak, by na wejściu każdego Raspberry Pi wynosiło właśnie 5 V.

Autor zamontował wtyczki 8P8N, popularnie określane mianem RJ-45, korzystając jedynie z 6 żył przewodów ethernetowych. Pozostałe dwie żyły wyprowadził obok, podłączając je z jednej strony do terminali śrubowych, do których dołączył kable biegnące z zasilacza. Z drugiej strony dolutowano wtyczki do goldpinów, które następnie można było wygodnie podłączać do komputerów Raspberry Pi. Zasilanie taśm diodowych zostało poprowadzone oddzielnie. W efekcie, udało się zredukować przewodów całej instalacji, co pozwoliło zarówno zmniejszyć jej koszty, jak i uprościć rozstawianie skanera.

Połączenie 40 komputerów do Ethernetu wymaga zastosowanie odpowiednio dużego switcha ethernetowego lub ich zestawu. Stosowny przełącznik można dobrać w zależności od liczby kamer. Na rynku łatwo dostępne są modele z kilkudziesięcioma

portami ethernetowymi oraz wersje zarządzalne, z portami typu „uplink”, które pozwalają na zestawianie ze sobą kilku switchy bez wprowadzania dodatkowych, istotnych opóźnień w transmisji. W praktyce nawet dla ponad 100 komputerów w pełni wystarczające będą switchy 100-megabitowe, przy czym warto, by łącze do komputera centralnego było gigabitowe. Nie ma potrzeby tworzenia sieci wirtualnych ani stosowania mechanizmów QoS, czy innych funkcji typowych dla przełączników zarządzalnych, ale szybkie, gigabitowe łącze do serwera lub pomiędzy switchami pozwoli znacznie szybciej wykonywać skan.

Naturalnie, moduły PI Camera zostały podłączone zgodnie z przeznaczeniem, a karty SD zostały umieszczone w czytnikach Raspberry Pi, po uprzednim nagraniu na nich oprogramowania.

### Konfiguracja komputerów

Autor skorzystał z systemu operacyjnego Raspbian, w wersji domyślnej, oferowanej bezpłatnie przez twórców Raspberry Pi. Wszystkie Raspberry Pi podłączone do sieci otrzymały stałe numery IP w ramach jednej podsieci. Autor uznał, że optymalnym rozwiązaniem będzie, by kamery wyzwalac zdalnie z centralnego komputera (laptopa) i by same zapisywały dane na centralnym serwerze – albo tym samym laptopie, albo innym komputerze czy dysku sieciowym. Oznacza to konieczność zastosowania odpowiedniego nazewnictwa plików ze zdjęciami, by można było rozpoznać, z której kamery pochodzi. Aby umożliwić wykonywanie kilku skanów pod rząd, każde kolejne zadanie skanowania skutkuje stworzeniem nowego katalogu na centralnym serwerze,



Rysunek 3. Model uchwytu kamery, drukowany na drukarce 3D

do którego komputery Raspberry Pi kopiują wykonane przez siebie zdjęcia.

Centralnie przechowywane są też ustawienia wszystkich komputerów Raspberry Pi, co znacząco ułatwia wprowadzanie wszelkich zmian w sposobie ich pracy. Skoro i tak muszą być podłączone do Ethernetu, by mogły wykonywać swoje zadanie i muszą korzystać z zewnętrznego serwera, by zapisywać na niego pliki, na tym samym serwerze można też przechowywać pliki konfiguracyjne. Domyślnym programem do wykonywania fotografii za pomocą modułu Pi Camera jest **raspistill**, którego plik konfiguracyjny `options.cfg` autor umieścił na serwerze. W efekcie uzyskuje się pewność, że wszystkie kamery będą wykonywać zdjęcia z dokładnie takimi samymi ustawieniami.

Pewne obawy mogłyby budzić synchronizacja wykonywanych zdjęć. Ethernet nie jest siecią deterministyczną czasowo i dopiero jego przemysłowe odmiany pozwalają zapewnić wysoką precyzję synchronizacji pracy urządzeń. W przypadku skanera 3D, w którym fotografowane są obiekty względnie nieruchome nie powinno to stanowić jednak wielkiego problemu, a dokładność na poziomie połowy sekundy powinna być

REKLAMA

Projekty na o.o.o

# STM32

www.stm32.eu

life.augmented

**KAMAMI**

wystarczająca. Fakt, że wszystkie komputery są tak samo ustawione i mają tę samą wydajność ułatwia wykonywanie zdjęć w tym samym czasie – pojawiające się opóźnienia będą identyczne dla wszystkich kamer. Znaczenie może mieć też zastosowanie dużych switchy ethernetowych, zamiast długiej drzewiastej struktury, w której droga sygnału od komputera sterującego do najdalszego Raspberry Pi jest znacznie dłuższa niż w do najbliższego. Jednakże kluczowe jest zastosowanie pakietów typu multicast, które w sieciach lokalnych są całkiem poprawnie obsługiwane. Dzięki nim, komputer sterujący przesyła do komputerów Raspberry Pi polecenie tylko raz, a zadaniem switchy ethernetowych jest rozesłanie ich do wszystkich kamer. Ponieważ switchy tworzą sobie mapę adresów IP dostępnych na poszczególnych swoich portach, błyskawicznie wiedzą, gdzie muszą przekazać otrzymany pakiet typu multicast i po prostu replikują go jednocześnie na wiele swoich wyprowadzeń. To sprawia, że wszystkie Raspberry Pi otrzymują polecenia jednocześnie, w efekcie czego autorowi udało się uzyskać synchronizację na poziomie poniżej 0,1 s.

## Wykonywanie zdjęć

Skrypty do kontroli komputerów Raspberry Pi zostały napisane w Pythonie. Wybór ten pozwala na ich maksymalne uproszczenie i wykorzystanie gotowych bibliotek oraz oprogramowanie zainstalowane w Raspbianie. Jeden skrypt Pythona jest uruchamiany na komputerach z kamerami, a inny – na komputerze sterującym. Całość działa na prostej zasadzie: Raspberry Pi nasłuchują na przychodzące komendy, analizują je i wykonują zadane polecenia, które mogą obejmować nie tylko wykonanie fotografii, ale też np. restart systemu. Skrypt komputera sterującego jest prostszy, gdyż po prostu przesyła podane polecenie do komputerów z kamerami, nie analizując go.

Skrypt sterujący komputerami z kamerami został przedstawiony na **listingu 1** i **listingu 2**. W tym przypadku jest to wersja pozbawiona funkcji restartowania systemu. Korzysta m.in. z biblioteki socket do prowadzenia komunikacji ethernetowej, subprocess do wywoływania zewnętrznych poleceń i sys do pobierania argumentów poleceń. W ogólności, uruchomienie skryptu powoduje włączenie nasłuchiwanie, na określonym porcie i wyzwolenie polecenia raspistill w momencie nadejścia komendy wykonania zdjęcia. Wykonane zdjęcie jest zapisywane na dysku sieciowym w oddzielnych katalogach dla każdego skanu.

Skrypt wyzwalający skanowanie jest jeszcze prostszy – po ustawieniu konfiguracji (numeru portu, grupy adresów multicast), sprawdzane są parametry wywołania skryptu i przesyłana jest multicastem komenda

### Listing 1. Nowa, uproszczona wersja skryptu uruchamianego na Raspberry PI z kamerami

```
#!/usr/bin/python
#Pi3Dscan - listen script
#Written by: Richard Garsthagen - richard@3dmij.nl
#More info: www.pi3dscan.com
#Version 2.0 - November 22nd 2014

import socket
import struct
import fcntl
import subprocess
import sys

MCAST_GRP = ,225.1.1.1'
MCAST_PORT = 3179

sock = socket.socket(socket.AF_INET, socket.SOCK_DGRAM, socket.IPPROTO_UDP)
sock.setsockopt(socket.SOL_SOCKET, socket.SO_REUSEADDR, 1)
sock.bind(('', MCAST_PORT))
mreq = struct.pack("4sl", socket.inet_aton(MCAST_GRP), socket.INADDR_ANY)
sock.setsockopt(socket.IPPROTO_IP, socket.IP_ADD_MEMBERSHIP, mreq)

print ""
print "3D Scanner - Open Source listen script"

debug = 1 # Turn debug message on/off

while True:
    data = sock.recv(10240)
    rdata = data[1:]
    rcmd = ord(data[0])
    if debug == 1:
        print "Received cmd: ," + str(rcmd)
        print "Data: " + rdata
    if (rcmd == 1):
        print "shooting"
        cmd = "raspistill " + rdata
        pid = subprocess.call(cmd, shell=True)
```

### Listing 2. Nowa wersja skryptu wysyłającego komendy wykonania skanu

```
#!/usr/bin/python
#Pi3Dscan - send script
#Written by: Richard Garsthagen - richard@3dmij.nl
#More info: www.pi3dscan.com
#Version 2.0 - November 22nd 2014

import socket
import struct
import fcntl
import subprocess
import sys
import IN

MCAST_GRP = ,225.1.1.1'
MCAST_PORT = 3179

if len(sys.argv) < 2:
    print "Please provide raspistill command options. See raspistill command
for details"
else:
    options = ""
    for a in range(1, len(sys.argv)):
        options = options + " " + sys.argv[a]
    print "Sending shooting command..."
    SCMD = chr(1) # Command 1 = Shoot photo
    SEND = SCMD + options
    sock = socket.socket(socket.AF_INET, socket.SOCK_DGRAM, socket.IPPROTO_
UDP)
    dev = "eth0" + "\0"
    sock.setsockopt(socket.SOL_SOCKET, IN.SO_BINDTODEVICE, dev)
    sock.setsockopt(socket.IPPROTO_IP, socket.IP_MULTICAST_TTL, 2)
    sock.sendto(SEND, (MCAST_GRP, MCAST_PORT))
    sock.close()
```

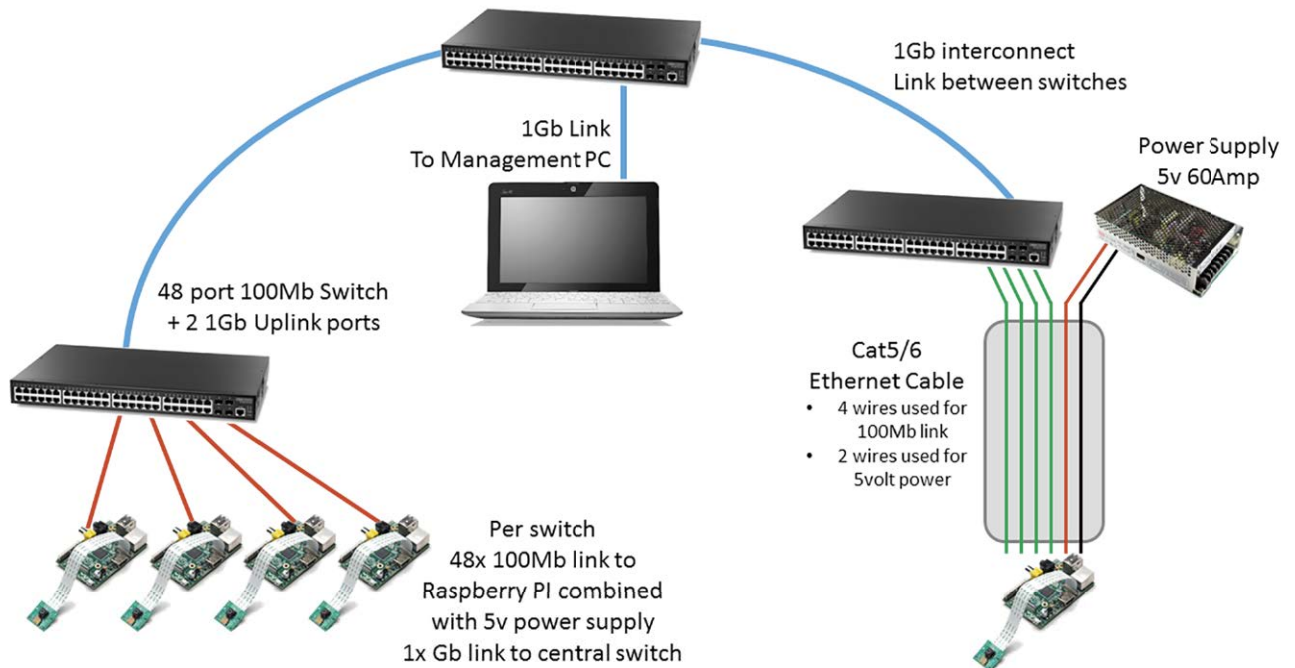
wykonania zdjęć. Następnie wystarczy zrobić zdjęcia tak, by powstał trójwymiarowy model.

## Konwersja zdjęć na model 3D

Z punktu widzenia elektronika, na opisanych dotąd działaniach praca nad projektem się kończy. Natomiast z punktu widzenia użytkownika, konieczne jest jeszcze wskazanie sposobu przetwarzania zgromadzonych zdjęć na trójwymiarowe modele. Nie jest to zadanie trywialne – właściwie jest najtrudniejszym elementem całego procesu, ale na rynku istnieje niemało gotowych rozwiązań, które w zautomatyzowany sposób tworzą modele 3D w oparciu o wgrane zdjęcia. Co więcej, niektóre z nich to narzędzia

przeznaczone dla początkujących użytkowników, dzięki czemu konwersja wymaga tylko kilku kliknięć. Jednakże, aby model był jak najbardziej dokładny, konieczne jest zrozumienie znaczenia niektórych dodatkowych opcji.

Dostępne jest zarówno proste, darmowe oprogramowanie, jak i bardziej zaawansowane, ale płatne. Większość pakietów jest także dostępnych w ograniczonych czasowo wersjach próbnych. Godne polecenia wydają się programy firm Autodesk (przede wszystkim profesjonalny RECAP, ale też amatorski 123D) oraz programy Agisoft Photoscan i VisualSFM. Alternatywą może być program Photo Modeler, czy nawet proste narzędzia on-line, takie jak Smoothie-3D. W zależności



Rysunek 4. Schemat połączenia komputerów

od rodzaju skanowanych obiektów, jedne z programów mogą okazać się bardziej adekwatne niż inne, przy czym oprogramowanie profesjonalne generalnie pozwala uzyskać lepsze rezultaty.

Konwersja jednego zestawu zdjęć na model 3D może długo trwać. Oczywiście zależy to od wydajności komputera, na którym się odbywa (czy dostępności chmury serwisu on-line), od wybranej precyzji modelu oraz od liczby i rozdzielczości wykonanych zdjęć. W praktyce będzie to często kilkadziesiąt minut na jeden model. Gotowy model najczęściej otrzymuje się w postaci pliku, który można podejrzeć albo w specjalistycznym oprogramowaniu, albo on-line w przeglądarce. Niekiedy może wystąpić konieczność ręcznego poprawienia modelu, poprzez połączenie ewentualnych luźnych punktów, których oprogramowaniu nie udało się

automatycznie zestawić ze sobą. Bardzo często konieczne będzie też ręczne usunięcie tła. Przydatne w tym celu może okazać się narzędzie Autodesk Memento.

### Wydruk gotowego modelu

Trójwymiarowe skanowanie można połączyć z inną, popularną obecnie techniką, tj. z drukiem 3D. Gotowy model można wydrukować, o ile tylko spełnia wymagania techniczne drukarek. Aby tak się stało, konieczne jest upewnienie się, że w modelu nie ma żadnych niepotrzebnych otworów i że wszystkie linie siatki modelu są ciągle tak, jak powinny być. Autor projektu również poleca do tego celu oprogramowanie Autodesk Memento, które wiele z tych operacji robi w sposób automatyzowany. Wystarczy by użytkownik skorzystał z prostego kreatora. Innymi przydatnymi programami, do poprawiania modelu

na potrzeby druku 3D mogą być bezpłatne aplikacje Pixologic Sculpttris i Autodesk Meshmixer. W końcu można też skorzystać z usług niezależnych firm, które oczyszczają modele 3D na zamówienie.

### Podsumowanie

Opisany projekt to niewątpliwie jedna z najciekawszych aplikacji z Raspberry Pi, jakie zrealizowano. Zdobyła już uznanie w wielu konkursach, organizowanych przez społeczność zajmującą się projektami z RPI. Jest to zarazem jeden z najdroższych w wykonaniu projektów, a przy zastosowaniu bardzo wielu kamer, jest także nużący. Przycinanie kilkudziesięciu kabli, montaż wtyczek na kablach i prowadzenie wszystkich połączeń to zadania żmudne i mało ciekawe, ale uzyskane rezultaty wydają się być godne poświęcenia. Sam koszt całej instalacji wynosi około 40-50 tysięcy złotych przy 100 kamerach. Naturalnie za największą część ceny



Rysunek 5. Oprogramowanie Autodesk Memento

REKLAMA

Projekty na...

# STM32

[www.stm32.eu](http://www.stm32.eu)

life.augmented



**Fotografia 6. Gotowy skaner z 115 kamerami oraz dodatkowymi 6 rzutnikami, które wyświetlają siatkę na fotografowanym obiekcie**

odpowiadają same komputery Raspberry Pi, ale koszt modułów Pi Camera ma podobne znaczenie.

Sam projekt jest ciekawy nie tylko ze względu na rezultaty, ale na zastosowane w nim technologie: Ethernet, zasilanie w stylu Power over Ethernet, przetwarzanie obrazów na model 3D oraz druk 3D, za którego pomocą wykonano uchwyty do montażu kamer. Ponadto autor przygotował zaawansowane oprogramowanie, które oferuje za opłatą, a które pozwala bardziej zautomatyzować wykonywanie dokładnych skanów, umożliwia zdalną obsługę skanera za pomocą pilota oraz optymalizuje sposób wykonywania zdjęć – każdy skan jest oparty o dwa zdjęcia wykonywane przez każdą kamerę, przy czym jedno z wyświetlona na skanowanym obiekcie siatką, która zwiększa precyzję odwzorowania powierzchni (funkcja dostępna tylko w połączeniu z oprogramowaniem Agisoft Photoscan).

Sukces projektu sprawił, że autor rozbudował go do 115 kamer i świadczy z jego użyciem usługi skanowania 3D „od ręki”. Sam skan trwa mniej niż sekundę, choć na gotowy model 3D, a tym bardziej na wydruk trójwymiarowej figurki trzeba trochę poczekać.

**Marcin Karbowniczek, EP**



**Fotografia 7. Figurki wydrukowane na drukarce 3D, powstałe w oparciu o trójwymiarowe modele wykonane skanerem**