

DOI: 10.5604/20830157.1121332

IMPLEMENTACJA OBSŁUGI PLIKÓW C3D W ŚRODOWISKU MATLAB

Grzegorz Koziel

Politechnika Lubelska, Instytut Informatyki, Zakład Ochrony Informacji

Streszczenie. Artykuł prezentuje zagadnienie dostępu do plików c3d w środowisku Matlab. Pliki c3d stanowią doskonały kontener do przechowywania danych ruchu wraz z zarejestrowanymi jednocześnie innymi danymi powiązаныmi z zarejestrowanym ruchem. Uzyskanie dostępu do danych zapisanych w pliku i prawidłowe ich przetworzenie stanowi podstawę obróbki tak przechowywanych danych. Artykuł przedstawia zastosowania formatu plików c3d oraz implementację procesu dostępu do danych ruchu w środowisku Matlab.

Słowa kluczowe: Matlab, c3d, Vicon, zapis ruchu

C3D FILE ACCESS IMPLEMENTATION IN MATLAB

Abstract. The subject of c3d files access in Matlab environment is presented in the article. The c3d files are a very good container to keep the movement data and other one related to the registered movement. Access to this data and its correct processing are the base of its analysis and usage. Applications of the c3d files and the access to the data contained in this file type in Matlab environment are presented in the article.

Keywords: Matlab, c3d, Vicon, motion record

Wstęp

Akwizycja ruchu to dziś bardzo pręźnie rozwijająca się dziedzina nauki. Akwizycja polega na zarejestrowaniu ruchu za pomocą kamer oraz zapisaniu pobranych w ten sposób danych na wybranym nośniku. Często razem z ruchem rejestrowane są również inne dane. Pozwala to na zwiększenie obszaru zastosowań akwizycji ruchu. Przykładem może być akwizycja ruchu idącego człowieka połączona z rejestracją nacisku stopy na podłoże czy też jednoczesna rejestracja pracy mięśni. Wszystkie zarejestrowane dane muszą zostać utrwalone by możliwa była ich późniejsza analiza. Co więcej nie mniej istotne jest poprawne zorganizowanie zapisu danych, tak by można było łatwo zidentyfikować, jaki element obiektu znajdował się w punkcie przestrzeni opisywanym przez pobrane dane oraz gdzie w tym samym momencie znajdowały się inne elementy obiektu. Istotna jest więc prawidłowa synchronizacja zapisywanych danych, uwzględniając oczywiście wszystkie dodatkowe dane rejestrowane przez dodatkowe czujniki [1, 5, 12]. Dobrym rozwiązaniem jest zastosowanie formatu pliku, który pozwoli na zapisanie wszystkich danych w jednym pliku. Pozwoli to uniknąć problemów z późniejszą synchronizacją ich podczas odczytu i analizy. Dodatkowo konieczne jest oznaczenie danych w taki sposób by możliwe było zidentyfikowanie, który punkt obiektu opisują oraz w którym momencie czasu. Wszystkie te wymagania spełnia format plików c3d. Dlatego jest on najczęściej wybieranym formatem zapisu danych ruchu wraz z powiązanymi z nimi danymi pochodzącymi z innych czujników [2, 4, 9, 10].

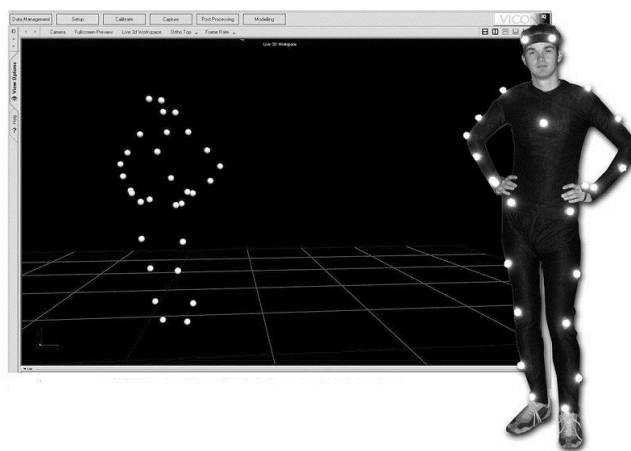
Kolejną kwestią, którą należy rozważyć jest wybór środowiska, w którym pliki te zostaną wczytane i poddane obróbce. W niniejszym artykule przedstawione zostało środowisko Matlab. Zostało ono wybrane ze względu na łatwość obsługi, gotowe biblioteki pozwalające przetwarzać i prezentować dane pochodzące z plików c3d.

1. Akwizycja danych ruchu

Dane ruchu pobierane są za pomocą kamer. Kamery rejestrują obraz, który następnie poddawany jest analizie. Jako, że niezbędne jest określenie położenia punktu w przestrzeni to obraz musi być rejestrowany jednocześnie przez wiele kamer. W wyniku otrzymywany jest więc zestaw obrazów tego samego obiektu zarejestrowany przez wiele urządzeń [3, 7]. Istotne w określaniu położenia określonego punktu w przestrzeni jest to aby został on zarejestrowany przez co najmniej dwie kamery w tej samej chwili czasu. W praktyce stosowane są więc systemy rejestrujące wyposażone w kilka lub nawet kilkanaście kamer. Im więcej kamer zarejestrowało położenie punktu w przestrzeni tym

precyzyjniej można określić jego położenie w przestrzeni. Nie bez znaczenia jest również fakt, że podczas rejestrowania ruchu złożonych obiektów może wystąpić zjawisko zasłaniania jednych obiektów przez inne. Wówczas część kamer nie będzie w stanie zarejestrować położenia zasłoniętych fragmentów obiektu. Nadal jednak, co najmniej dwie kamery muszą rejestrować obraz by możliwe było określenie położenia analizowanego punktu. Należy jednak wspomnieć, że określanie położenia punktu w przestrzeni trójwymiarowej przy pomocy dwóch kamer możliwe jest tylko wtedy, gdy znane jest dokładne umiejscowienie kamer w przestrzeni oraz ich orientacja [7].

Kolejnym aspektem istotnym podczas akwizycji ruchu jest precyzyjne określenie położenia danego fragmentu obiektu na obrazie zarejestrowanym przez różne kamery. Rozpoznawanie obrazu jest nadal bardzo złożoną i trudną dziedziną. Problemy dotyczą zwłaszcza obrazów, na których brak jest charakterystycznych punktów lub krawędzi. Stosowanym rozwiązaniem tego problemu jest umieszczenie na obiekcie specjalnych markerów wyznaczających położenie punktów charakterystycznych dla danego obiektu. Znaczniki te umieszczane są w punktach, których umiejscowienie jest istotne z punktu widzenia rejestrowanego ruchu. Rejestracja położenia tych punktów pozwala na łatwe określenie położenia obiektu w przestrzeni trójwymiarowej [13]. Przykładowy wygląd sylwetki człowieka z zamocowanymi markerami oraz odwzorowanie położenia markerów w programie do analizy danych przedstawione zostały na rysunku 1.



Rys. 1. Sylwetka człowieka wraz z zamocowanymi markerami oraz jej reprezentacja w programie komputerowym [13]

2. Format plików c3d i jego zastosowanie

Format pliku c3d (ang. Coordinate 3D) zapewnia możliwość zapisywania współrzędnych w przestrzeni trójwymiarowej oraz powiązanych z nimi danych cyfrowych i analogowych wraz z ich parametrami. Pierwotnie był zaprojektowany do obsługi fotogrametrycznego systemu informatycznego AMASS. Miał umożliwiać zapisanie danych kalibracji, śledzenia, identyfikacji znaczników oraz innych danych niezbędnych do prawidłowego odczytania zawartych w nim danych. Miał również zapewniać możliwość zapisania ścieżek 3D oraz danych analogowych. Głównymi priorytetami przyjętymi podczas projektowania formatu c3d były [6]:

- zapewnienie możliwości gromadzenia różnego typu danych wewnątrz pliku,
- łatwe zapisywanie parametrów oraz ich wartości w sekcji „parametry”,
- umożliwienie stosowania opisowych nazw parametrów oraz dołączania dodatkowych opisów, tak by plik był samo komentujący się,
- dostarczenie użytkownikom narzędzia, które umożliwiało by dodawanie, sprawdzanie i modyfikację dowolnego parametru wewnątrz pliku,
- zaprojektowanie wydajnego i spójnego magazynu mogącego zawierać wszystkie niezbędne informacje wewnątrz jednego pliku.

Format pliku c3d definiuje składnię pliku zawierającą trzy główne komponenty:

- 1) dane – komponent przeznaczony do przechowywania w formie binarnej danych 3D oraz informacji analogowej,
- 2) parametry standardowe – informacje o danych 3D oraz analogowych niezbędne do ich poprawnego odczytania,
- 3) parametry dodatkowe – dodatkowe parametry dodawane podczas akwizycji danych lub w dowolnym momencie na potrzeby ich przetwarzania i gromadzenia.

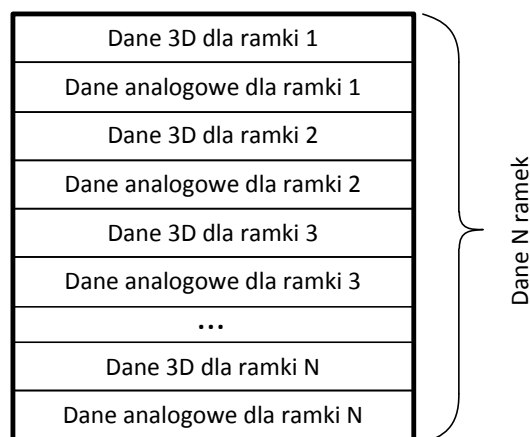
Dane w pliku c3d przechowywane są w postaci szesnastobitowej liczby całkowitej ze znakiem lub w postaci zmiennoprzecinkowej. Format danych definiowany jest poprzez wartość zawartą w nagłówku pliku. Zdefiniowany format danych dotyczy wszystkich wartości zapisanych w pliku – zarówno cyfrowych jak i analogowych. Nie ma możliwości zastosowania jednego typu danych do zapisywania danych analogowych a innego typu danych do opisanie danych cyfrowych. Dodatkowo w celu zachowania kompatybilności z językiem Fortran oraz niektórymi systemami operacyjnymi dane w plikach c3d powinny być zorganizowane w bloki o rozmiarze 512 lub 256 bajtów.

Plik w formacie c3d musi więc zawierać co najmniej trzy sekcje:

- 1) nagłówek – pojedyncza sekcja o rozmiarze 512 bajtów,
- 2) sekcję parametrów zawierającą jeden lub więcej 512 bajtowych bloków; sekcja parametrów powinna znaleźć się w każdym pliku c3d; jej rozmiar jest zmienny, typowo zawiera od ośmiu do dziesięciu bloków,
- 3) sekcję danych zawierającą dane trójwymiarowe oraz powiązane z nimi dane analogowe umieszczone w jednym lub większej liczbie 512 bajtowych bloków; sekcja ta umieszczona jest bezpośrednio za sekcją parametrów; jej początek wskazuje znacznik POINT:DATA_START umieszczony w sekcji parametrów.

Akwizycja danych ruchu odbywa się za pomocą wielu urządzeń. Są to zarówno kamery jak i inne urządzenia rejestrujące dodatkowe wartości mierzone podczas ruchu obiektu. Ponadto dane ruchu to zapis położenia wielu markerów w przestrzeni. Charakterystyczne jest to, że wszystkie te dane są pobierane w jednej dyskretnej chwili czasu. Oznacza to, że również zapis powinien być zorganizowany w taki sposób by wszystkie dane pobrane w tym samym momencie czasu zostały zapisane w sposób umożliwiający określenie czasu rejestracji tej grupy danych. W plikach c3d problem ten został rozwiązany poprzez wprowadzenie ramek. Jest to konstrukcja analogiczna do ramek występujących w animacji. Ramka określa chwilę czasu, w której

zarejestrowane zostały dane. Wewnątrz niej umieszczane są wszystkie wartości zarejestrowanych danych. Dodatkowo w pliku c3d naprzemiennie umieszczane są dane określające położenie punktów w przestrzeni trójwymiarowej oraz powiązane z nimi dane analogowe. Struktura danych zapisanych w pliku c3d przedstawiona została na rysunku 2.



Rys. 2. Struktura danych zapisanych w pliku c3d [6]

3. C3Dserver

Środowisko Matlab nie zapewnia bezpośredniego wsparcia obsługi plików c3d. Konieczne jest wykorzystanie dodatkowego narzędzia pośredniczącego pomiędzy plikiem a Matlabem. Narzędzie to obsługuje pliki c3d przekazując odczytane dane do Matlaba, gdzie następuje ich analiza i obróbka.

Jedną z możliwości jest zastosowanie oprogramowania C3Dserver. Jest to dodatkowy program wykonujący żądane operacje na plikach c3d. C3Dserver udostępnia innym programom interfejs COM pozwalający na komunikację i wymianę danych. Aby możliwe było korzystanie z interfejsu COM programu C3Dserver konieczne jest jego wcześniejsze aktywowanie. Dokonuje się tego za pomocą polecenia `itf = c3dserver()`. Polecenie to jako wynik działania zwraca obiekt COM utworzonego serwera, który w przedstawionym przykładzie został zapisany pod nazwą `itf`.

Aktywowanie interfejsu COM pozwala na rozpoczęcie pracy z plikami c3d. Możliwe jest otwarcie pliku. Dokonuje się tego za pomocą polecenia `openc3d(itf)`. Parametrem polecenia jest obiekt utworzonego wcześniej interfejsu COM. Wywołanie tego polecenia powoduje utworzenia standardowego okienka wyboru pliku do otwarcia. Wybranie i zaakceptowanie pliku umożliwia dostęp do jego zawartości [6].

Poprzez utworzony interfejs COM możliwe jest odczytanie danych. Struktura fizyczna danych ukryta jest przed użytkownikiem. Nie występuje podział na ramki. Natomiast interfejs wyraźnie dzieli dane na serie. Każda z serii zawiera wszystkie dane pochodzące z jednego czujnika pomiarowego lub dotyczące położenia jednego z markerów. Problematyczna – przynajmniej w początkowym etapie pracy – jest konieczność samodzielnego wprowadzenia nazw znaczników występujących w pliku.

Interfejs nie dostarcza możliwości automatycznego odczytania listy nazw znaczników. Jeżeli dane pochodzą zawsze z tego samego laboratorium, dającego każdorazowo plik o takiej samej strukturze znaczników, brak możliwości automatycznego odczytu listy znaczników zapisanych w pliku nie stanowi większego problemu. Jednak w przypadku obróbki plików zawierających różne znaczniki problem staje się poważny i konieczny jest większy nakład pracy oraz dodatkowe oprogramowanie pozwalające na odczytanie listy znaczników, którą należy następnie przenieść do środowiska Matlab w celu umożliwienia poprawnego odczytu plików. Listę znaczników umieszczamy w tablicy lub dowolnej innej strukturze pozwalającej na łatwe jej odczytanie i określenie liczby elementów. Przykładowa tablica

znaczników ma postać: `markers_names={'RFT1'; 'RFT2'; 'LFT1'}`.

Na podstawie nazw znaczników możliwe jest odczytanie list wartości przypisanych do poszczególnych znaczników. Każda lista zawiera kolejne wartości (położenia) zmierzone w następujących po sobie ramkach. Wczytanie danych z wybranej serii możliwe jest poprzez wywołanie polecenia `get3dtarget`.

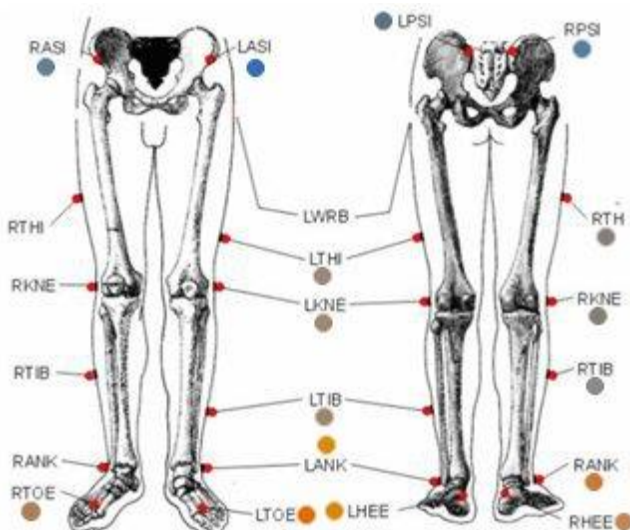
Polecenia to wymaga podania co najmniej dwóch parametrów: zmiennej reprezentującej obiekt COM realizujący dostęp do pliku oraz nazwy znacznika powiązanego z odczytywaną serią danych. Polecenie wywołane z dwoma argumentami zwraca dwuwymiarową tablicę o rozmiarze $N \times 3$, gdzie N oznacza liczbę ramek. Dla każdej ramki zwracane są trzy liczby określające położenie względem osi X, Y oraz Z. Dodatkowo możliwe jest podanie dodatkowych parametrów do polecenia `get3dtarget`. Podanie czwartego argumentu powoduje, że zwrócona zostanie tablica $N \times 4$. Pierwsze trzy wartości, dla każdej ramki nie zmienia się. Czwarta będzie zawierała informację o residuum.

Podanie dodatkowych dwóch argumentów na pozycji czwartej i piątej pozwala na podanie początkowej oraz końcowej ramki odczytywanych danych. Pozwala to ograniczyć ilość odczytywanych danych do żądanego fragmentu. W przypadku określania początkowej i końcowej ramki wymagane jest podanie również trzeciego parametru. Możliwe jest jednak jego pominięcie przez podanie wartości zero. Ogólna składnia polecenia ma postać: `XYZPOS = get3dtarget(itf, signalname, residual, index1, index2)`. Macierz XYZPOS jest macierzą, w której zostaną zapisane wyniki. Współrzędne X, Y, Z, oraz residuum umieszczone zostaną w kolejnych kolumnach macierzy [6].

4. Prezentacja przykładowych danych

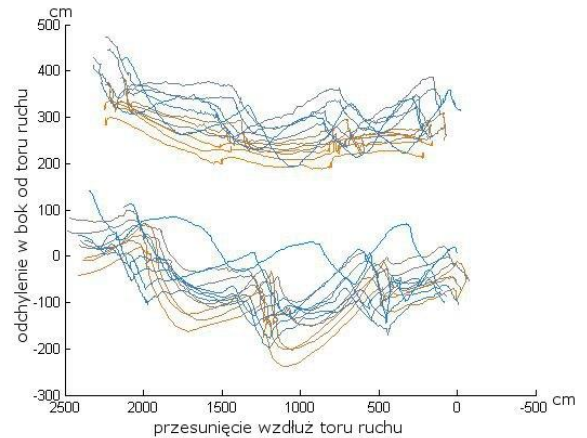
W celu przetestowania działania opisanego w poprzednich rozdziałach rozwiązania pozwalającego na wczytywanie danych ruchu opracowany został program wczytujący zawartość pliku z danymi oraz wizualizujący na trójwymiarowym wykresie ruch wszystkich markerów. Prezentowany w niniejszym artykule przykład jest wizualizacją ruchu człowieka idącego na wprost. Wizualizacja dotyczy dolnej połowy ciała od bioder w dół. Pokazuje więc ruch bioder oraz nóg w trakcie chodu. Na rysunkach 4, 5 i 6 przedstawione zostały trzy rzuty wygenerowanego wykresu.

Dane ruchu do utworzenia wykresu udostępniło Laboratorium Akwizycji Ruchu Polsko-Japońskiej Wyższej Szkoły Techniki komputerowych. Rejestracja ruchu obejmowała określenie pozycji markerów umiejscowionych na ciele człowieka. Umieszczenie markerów prezentuje rysunek 3.

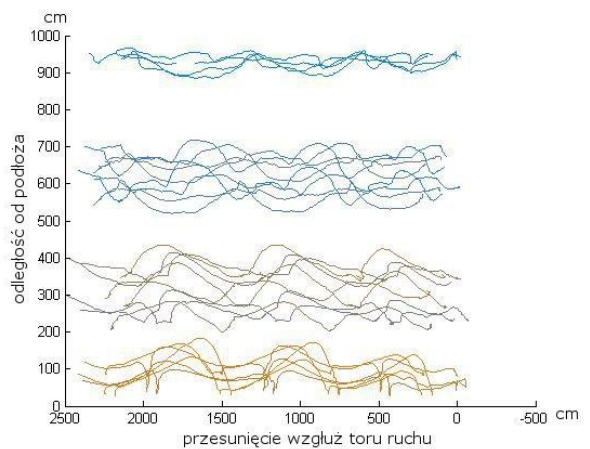


Rys. 3. Umieszczenie znaczników ruchu na ciele człowieka [8]

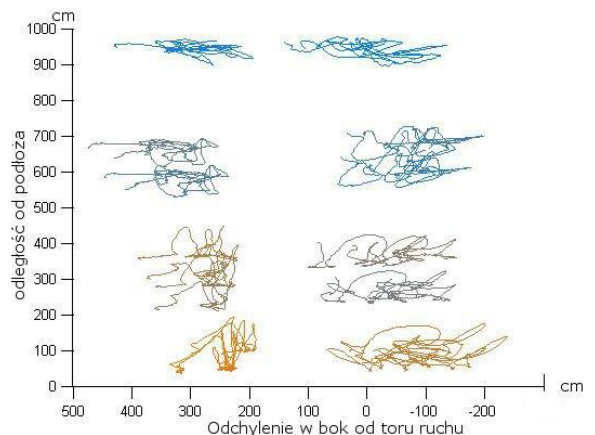
Na rysunkach 4-6 możemy zauważyć kolorowe krzywe. Są to przebiegi pokazujące tor ruchu markerów umieszczonych na ciele człowieka. Każdemu znacznikowi został przypisany inny kolor na prezentowanych rysunkach. Dla ułatwienia analizy na rysunku 3 obok oznaczenia każdego znacznika został umieszczony kolor odpowiadający mu na rysunkach 4-6. Analiza rysunku 4 pokazuje jak wyglądał ruch poszczególnych części ciała w rzucie z góry. Można zauważyć, że linie tworzą dwie grupy. Spowodowane jest to umieszczeniem znaczników po bokach ciała. Tak więc jedną grupę linii tworzą tory ruchu znaczników umiejscowionych po prawej stronie ciała a drugą grupę tory znaczników umiejscowionych po lewej stronie ciała. Jak łatwo zauważyć chód człowieka nie jest idealnie równy i charakteryzuje się chwiejnością i przemieszczaniem ciała człowieka na boki.



Rys. 4. Wizualizacja danych ruchu człowieka – rzut z góry



Rys. 5. Wizualizacja danych ruchu człowieka – rzut z boku



Rys. 6. Wizualizacja danych ruchu człowieka – rzut z tyłu

Rysunek 5 prezentuje widok z boku na wykresy ruchu poszczególnych markerów. Możemy tam zauważyć cztery grupy linii. Najniższa grupa to wizualizacja ruchu znaczników umiejscowionych na stopach. Dwie środkowe grupy odpowiadają markerom umieszczonym na nogach powyżej stóp. Najwyższa grupa to znaczniki obrazujące ruch bioder człowieka. Analizując wykres przedstawiony na rysunku 5 możemy stwierdzić, że przedstawiony ruch charakteryzuje się pewnymi prawidłowościami: najmniejsze wahania wykazuje ruch bioder, które jedynie w niewielkim stopniu przemieszczają się w górę i w dół.

Ruch nóg charakteryzuje się znacznie większym zróżnicowaniem wysokości położenia znaczników w czasie. Jest to spowodowane unoszeniem nóg podczas chodu. Na najniższych seriach danych można zauważyć charakterystyczny powtarzający się wzór: krótkie momenty, w których wykres osiąga minimum oraz dłuższe, w których wysokość znacznika osiąga pewien zbliżony do stałego poziom. Minima odpowiadają chwilom, w których stopa styka się z podłożem. Dłuższe fragmenty odpowiadają fazie przenoszenia nogi podczas wykonywania kroku.

Rysunek 6 przedstawia prezentację danych ruchu w rzucie z tyłu. Możemy na nim określić zakres zmienności położenia markerów w pionie i w poziomie. Daje się również zauważyć dosyć duży stopień podobieństwa w wykresach obrazujących ruch poszczególnych markerów. Świadczy to o koordynacji ruchu, jako całości.

C3D Server w połączeniu z Matlabem pozwala uniknąć wyświetlania błędnych danych dla punktów, dla których pomiar nie został wykonany. W większości rozwiązań punkty te przyjmują błędną wartość, co zniekształca wyświetlaną trajektorię ruchu.

Wykonanie poprawnej trajektorii wymaga detekcji błędnych wartości oraz interpolacji sygnału [11]. W Matlabie problem ten nie występuje, jako że wartość tego typu punktów reprezentowana jest jako NaN (Not a Number), co powoduje, że nie są one wyświetlane na wykresach. W miejscu błędnego punktu trajektoria ruchu nie jest wyświetlana.

5. Wnioski

Pliki c3d są bardzo wygodnym i wydajnym narzędziem do przechowywania danych z systemu akwizycji ruchu. Zwłaszcza jeżeli dane te są powiązane z innymi pochodzącymi z niezależnych systemów pomiarowych, które muszą jednak być zsynchronizowane z danymi ruchu.

Matlab jako środowisko nie wspiera bezpośrednio obsługi plików c3d. Możliwy jest jednak dostęp do tego typu plików przy zastosowaniu dodatkowych narzędzi, czego przykładem jest zaprezentowany w artykule interfejs COM C3DServer. Matlab dostarcza możliwości szybkiej i precyzyjnej obróbki, analizy i wizualizacji danych ruchu.

Interfejs C3DServer jest użyteczny i pozwala na wygodną pracę z danymi zawartymi w plikach c3d. Narzuca jednak pewne ograniczenia przez to, że nie oferuje pełnych możliwości operowania na plikach c3d czego przykładem może być brak możliwości odczytania nazw znaczników zapisanych w pliku.

Podziękowania

Autor jest uczestnikiem projektu "Kwalifikacje dla rynku pracy - Politechnika Lubelska przyjazna dla pracodawcy" współfinansowanego przez Unię Europejską w ramach Europejskiego Funduszu Społecznego.

Literatura

- [1] Choong-Hoon Kwak, Bajic I.V.: Hybrid low-delay compression of motion capture data. 2011 IEEE International Conference on Multimedia and Expo pp. 1-6.
- [2] Choong Seng Chan, Tsai F. S.: Computer Animation of Facial Emotions. 2010 International Conference on Cyberworlds, pp.425-429.
- [3] Gleicher M., Ferrier N.: Evaluating Video-Based Motion Capture, Proceedings of Computer Animation 2002.
- [4] Łukasik E., Skublewska-Paszowska M.: E-learning as a modern tool in corporation training, Actual Problems of Economics (in press) Świtez, 2012.
- [5] Moelsund T. B., Granum E.: A Survey of Computer Vision-Based Human Motion Capture. Computer Vision and Image Understanding, 2001, pp. 231-268.
- [6] Motion Lab Systems, The C3D File Format User Guide.
- [7] Parent R.: Animacja komputerowa, PWN 2011.
- [8] Polsko-Japońska Wyższa Szkoła Technik Komputerowych, Prezentacja zebranych danych Ruchowych, Bytom 2010.
- [9] RuiHu Kuang, JuChi Kuang: Structuring of Experiential Learning Platform of Virtual-reality Commodity for Remote E-Business. 2010 International Conference on E-Business and E-Government, pp. 4550-4553.
- [10] Said Hamdioui, Mottaqjallah Taouil: Yield Improvement and Test Cost Optimization for 3D Stacked ICs. 2011 Asian Test Symposium, pp. 480-485.
- [11] Skublewska-Paszowska M., Łukasik E., Smolka J.: Analysis of motion interpolation methods, in Actual Problems of Economy, Shatsk Lakes, Ukraine, July 2-6, 2012, in press.
- [12] Suwichai Phunsa, Nawuttagorn Potisarn, Suwich Tirakoat: Edutainment -- Thai Art of Self-Defense and Boxing by Motion Capture Technique. 2009 International Conference on Computer Modeling and Simulation, pp. 152-155.
- [13] explosion.com/motion_capture
- [14] www.galileosworld.com
- [15] skyreignsoftware.com/resourcepage/low-cost-motion-capture-technology

Dr inż. Grzegorz Koziel
e-mail: g.koziel@pollub.pl

Absolwent Wydziału Elektrycznego Politechniki Lubelskiej, gdzie również uzyskał tytuł doktora w 2011 roku, broniąc pracę „Zmodyfikowane metody cyfrowego przetwarzania sygnałów dźwiękowych w steganografii komputerowej”. Pracownik Instytutu Informatyki na Wydziale Elektrotechniki i Informatyki Politechniki Lubelskiej, początkowo jako asystent, później jako adiunkt a obecnie zastępca dyrektora Instytutu ds. nauki. Działalność naukowa obejmuje steganografię oraz analizę danych ruchu.

