

Generowanie przestrzennej trajektorii robota chirurgicznego za pomocą półprzewodnikowego sensora ruchu o 6 stopniach swobody

Ryszard Leniowski

Katedra Informatyki i Automatyki, Politechnika Rzeszowska

Streszczenie: Praca prezentuje alternatywne podejście do konstrukcji zadajnika trajektorii robota chirurgicznego, polegające na wykorzystaniu sprzężonych sensorów ruchu typu MEMS: kompasu, żyroskopu, akcelerometru. Miniaturowy moduł z tymi układami może być nakładany bezpośrednio na dłoń chirurga-operatora lub na jego palce. Zaletą takiego zadajnika jest nie tylko niska cena, ale również wysoka funkcjonalność, związana z otwartością na dodawanie nowych modułów programowych. Przykładem wbudowanej funkcji może być generator ścieżki dojścia do pola zabiegu, wykorzystujący wielomiany sklepane w reprezentacji *Catmull-Rom*. Dodatkowo, zadajnik tego typu jest bardzo mały i łatwy do zastosowania w systemie mobilnym (karetka pogotowia, polowy ambulans wojskowy).

Słowa kluczowe: robotyka, roboty medyczne, planowanie trajektorii ruchu, sensory ruchu

1. Wprowadzenie

Trajektoria ruchu robota chirurgicznego jest generowana na bieżąco przez chirurga-operatora za pomocą zadajników ruchu. Najczęściej są to urządzenia elektromechaniczne o skomplikowanej budowie. Opis przykładowych konstrukcji zawiera [1]. Zadajniki jako konstrukcje mechaniczne, pomimo wielu zalet, posiadają też wady, takie jak: znaczne gabaryty, ciężar i wysoka cena. Najczęściej są to obudowane konsole, wykonane jako urządzenia stacjonarne.

Proponowane w pracy alternatywne podejście do konstrukcji zadajnika trajektorii polega na wykorzystaniu sprzężonych sensorów ruchu typu MEMS [2, 3]. Miniaturowy moduł z tymi układami może być nakładany bezpośrednio na dłoń chirurga-operatora lub na wybrane palce (wykorzystywanych jest wtedy kilka modułów). W razie potrzeby dodatkowy moduł można przypiąć do ramienia operatora, tak aby jak najpełniej zarejestrować wszystkie ruchy ręki. Zaletą proponowanego zadajnika jest nie tylko niska cena, ale również otwartość na nowe funkcje, które można dodawać wraz z nowym oprogramowaniem. Ponadto, zadajnik tego typu jest bardzo mały i łatwy do zastosowania w systemie mobilnym (karetka pogotowia, polowy ambulans wojskowy). Wadą rozwiązania jest trudniejsza (ale możliwa) realizacja toru reakcji siłowej, tzw. *force feedback*.

Zasadniczą częścią zadajnika są dwa czujniki MEMS, realizujące sprzężony pomiar 12 parametrów ruchu, czyli: prędkości liniowych i pozycji w trzech wymiarach oraz

prędkości kątowych i orientacji, również w trzech wymiarach. Pełny wektor parametrów ruchu pozwala na sterowanie robotem chirurgicznym ROCH-1 [4] i realizację trajektorii prosto- i krzywoliniowych, składających się na czynności takie, jak: trzymanie (klips), cięcie (skalpel), szycie, nakłuwanie (igła). Faza ruchu, odpowiadająca za przejście przez trokar (rodzaj przewodnicy rurowej w skórze pacjenta), jest realizowana półautomatycznie. Operator uruchamia komendy logiczne (ruch lub jego wstrzymanie). Może oczywiście korygować trajektorię wyznaczoną przed zabiegiem na podstawie przetworzonych danych, otrzymanych ze skanowanego fragmentu ciała (mapa powierzchni) i obrazów CT [5].

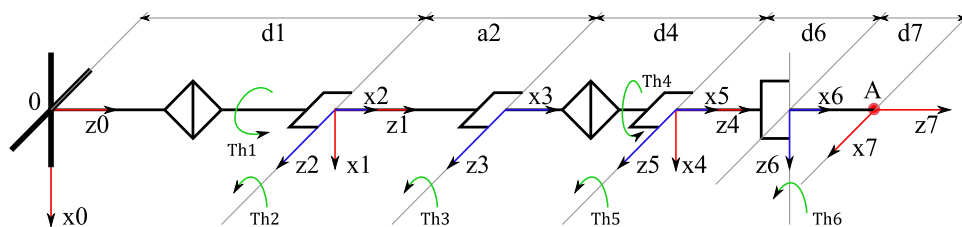
2. Charakterystyka ruchów manipulacyjnych robota chirurgicznego ROCH-1

Pokazany na rys. 1 robot chirurgiczny ROCH-1 jest mocowany do robota-matki, który przenosi go w pobliże pola operacyjnego. ROCH-1 zbudowany jest z sześciu modułów o zróżnicowanych długościach i średnicy poniżej 10 mm, połączonych przegubami obrotowymi. Człony zawierają napędy, sensory ruchu, sterowniki (1 moduł) oraz okablowanie. Cztery przeguby mają wbudowane wielostopniowe przekładnie planetarne i ślimakowe. W dwóch przegubach zastosowano napęd bezpośredni z czterema siłownikami piezoelektrycznymi.



Rys. 1. Robot chirurgiczny ROCH-1
Fig. 1. Surgical robot ROCH-1

Robot ROCH-1 w konfiguracji z rys. 2 ma 6 stopni swobody o układzie kinematycznym 6R. Parametry D-H zawarto w tab. 1. Bazowy układ odniesienia dla robota ROCH-1, oznaczony literą O, przyjmuje takie położenie i taką orientację względem ciała pacjenta, jaka wynika z rodzaju zabiegu.

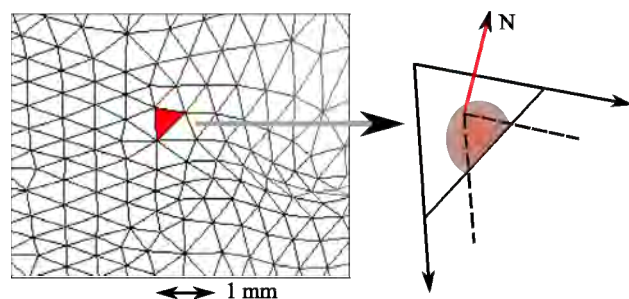


Rys. 2. Układ kinematyczny robota chirurgicznego ROCH-1
Fig. 2. Systems of coordinates of surgical robot ROCH-1

Tab. 1. Parametry kinematyczne robota ROCH-1
Tab. 1. Kinematic parameters of robot ROCH-1

Numer	a_{i-1}	α_{i-1}	d_i	θ_i	Zmienna	Zakres
1	0	0	d_1	0	θ_1	
2	0	90	0	90	θ_2	
3	a_2	0	0	0	θ_3	
4	0	-90	d_4	0	θ_4	
5	0	90	0	90	θ_5	
6	a_5	90	0	90	θ_6	
7	0	-90	d_7	0	-	

Naprowadzanie robota ROCH-1 na trokar wykonuje robot-matka. Miejsce trokaru (pozycja) oraz orientację płaszczyzny wejścia do wnętrza ciała pacjenta ustala się na podstawie obrazów CT oraz mapy fragmentu ciała wykonanej skanerem 3D (rys. 3). Uzyskiwana siatka skanowanej powierzchni jest trójkątna i może być wyznaczana z dokładnością lepszą niż 0,5 mm. W ustalonym punkcie wprowadzenia trokaru (oznaczono wyróżnionym kolorem), na podstawie wierzchołków trójkąta, oblicza się normalną. Wyznacza ona kierunek wejścia do wnętrza ciała i jednocześnie orientację płaszczyzny (trójkąta) w układzie globalnym.



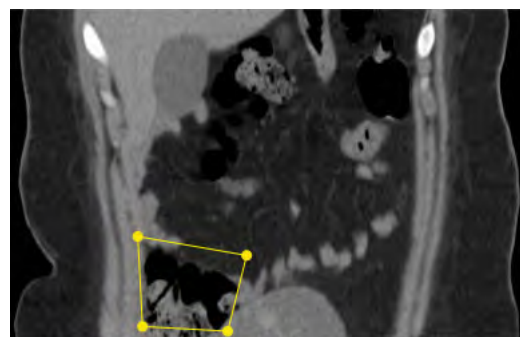
Rys. 3. Wyznaczenie orientacji płaszczyzny wejścia do wnętrza ciała
Fig. 3. Orientation determination of the entrance plane to the body interior

Faza przejścia przez trokar jest realizowana półautomatycznie, w czterech etapach. Wykorzystuje się przy tym układy równań kinematyki manipulatora, z tzw. punktem

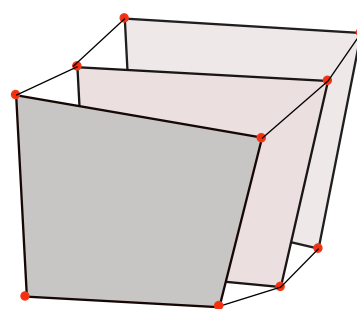
stałym. Szczegółowy opis tego procesu został zaprezentowany w [6]. Robot porusza się w wytworzonej na potrzeby zabiegu przestrzeni, powstałej przez wprowadzenie do organizmu pacjenta kontrolowanej ilości obojętnego gazu. Kanał dojścia do pola operacyjnego wyznacza się na podstawie zbioru indeksowanych zdjęć z CT, które są

przekrojami pacjenta, przesuniętymi względem siebie o 0,5 mm lub o 1 mm. Na każde kolejne zdjęcie nanosi się czworokąt wypukły. Wierzchołki o tych samych numerach dla kolejnych przekrojów

łączy się, (rys. 4a). Powstaje w ten sposób kanał o zmiennym czworokątnym przekroju (rys. 4b). Ponieważ wierzchołki kolejnych przekrojów tworzących ściany boczne nie leżą na jednej płaszczyźnie, wypełnia się je dwoma trójkątami (rys. 5a).



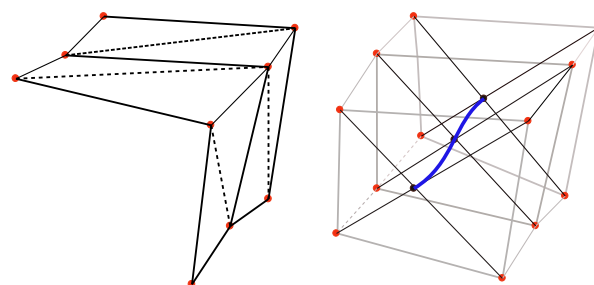
(a)



(b)

Rys. 4. Nanoszenie wierzchołków na obraz CT (a), czworokąt wypukły definiujący kanał ruchu (b)

Fig. 4. Inserting vertices to the CT image (a), convex quadrilaterals defining the traffic channel (b)



(a)

(b)

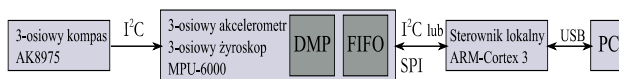
Rys. 5. Wypełnianie boków kanału (a), trajektoria przejścia (b)
Fig. 5. Fill of the channel sides (a), the crossing trajectory (b)

Przejsie do obszaru zabiegu odbywa sie po trajektorii wyznaczonej przez punkty przeciecia przekatnych kolejnych czworokatow wypuklych (rys. 5d). Tworza one wektor punktow kontrolnych kubicznych wielomianow sklepanych. Problem ten omowiono dokladniej w punkcie czwartym. Po dotarciu do pola zabiegu za pomoca zadajnika z polprzewodnikowymi sensorami ruchu nastepuje przełączenie trybu pracy na sterowanie ręczne. Szczegoly budowy tego urzadzenia, proces pozyskiwania danych oraz wyznaczenie na ich podstawie parametrów ruchu sa prezentowane w kolejnych punktach pracy.

3. Projekt zadajnika trajektorii ruchów robota ROCH-1

Projekt zadajnika trajektorii ruchów robota ROCH-1 zostal oparty na dwuch ukladach – sensorach typu MEMS, tj. MPU-6000 firmy IvenSense Inc., oraz AK8975 firmy Ashai KASEI. Pierwszy z wymienionych ukladów jest zintegrowanym akcelerometrem i żyroskopem wyznaczajacym przyspieszenie i predkosci katowe w trzech osiach lokalnego ukladu wspolrzednych. Posiada wbudowany blok wstepnego przetwarzania danych na poziomie sprzetowym, tzw. DMP (ang. *Digital Motion Processor*). Jego rola polega na przetwarzaniu danych z wewnetrznych przetworników A/D, zapisaniu ich w rejestrach danych oraz na obsludze interfejsów SPI i I²C. Poprzez te dwa porty komunikacyjne uklad wymienia dane z kontrolerem nadrzednym. Jeden z portów mozna wykorzystac do podlaczzenia dodatkowego sensora, który posiada port I²C.

W proponowanym rozwiazaniu nie skorzystano z tej mozliwosci. Wprowadzajac do zadajnika drugi sensor, tj. AKM8975, podlaczono go bezposrednio do sterownika lokalnego. Jest to czujnik pola magnetycznego Ziemi czyli kompas. Pełni on role generatora referencyjnych orientacji zadajnika w przestrzeni trójwymiarowej. Podlaczajac kompas AK8975 bezposrednio do MPU-6000, otrzymujemy mozliwosc rejestracji 9 danych: 3 przyspieszen linowych, 3 predkosci katowych, 3 katów orientacji zadajnika względem Ziemi. Ideę takiego rozwiazania z kontrolerem nadrzednym, wyposazonym w procesor ARM, przedstawiono na rys. 6. Zaznaczony na rysunku rejestr FIFO ma pojemnosc 1024 bajtów. W takiej formie zadajnik trajektorii jest postrzegany przez uklad nadrzedny (kontroler ARM).



Rys. 6. Idea pomiaru parametrów ruchu
Fig. 6. The idea of measuring traffic parameters

Stosowane uklady pomiarowe cechuja sie wysokimi parametrami. Najwazniejsze z nich podano w tab. 2 i 3.

Pamiatajac, ze natężenie pola magnetycznego na powierzchni Ziemi wynosi okolo 50 μT widzimy, ze uklad AK8975 moze wyznaczyć kierunek pola z dokladnoscia nie gorsza niz 0,6 %. Oba uklady sa wykonane w formacie 4 × 4 mm, co pozwala na skonstruowanie zadajnika o wymiarach umozliwiajacych jego mocowanie do palców ręki operatora.

Tab. 2. Wybrane parametry ukladu MPU-6000

Tab. 2. Selected parameters of MPU-6000

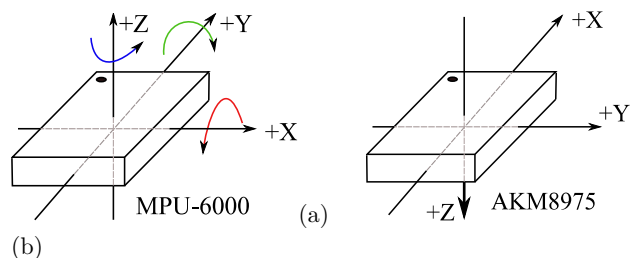
Parametr	Warunki	Wartosc
Zakres pomiaru predkosci katowej	25 °C	±2000°/s ±1000°/s ±500°/s ±250°/s
Czulość pomiaru predkosci katowej	±2000°/s ±1000°/s ±500°/s ±250°/s	16.4 LSB/(°/s) 32.8 LSB/(°/s) 65.5 LSB/(°/s) 131 LSB/(°/s)
Pasma pomiarowe dla osi X, Y, Z		36, 33, 30 kHz
Nieliniowosc		0,2 %
Separacja miedzyosiowa		±2 %
Rozdzielczosc pomiarowa		16 bitów
Zakres pomiaru przyspieszenia		±2g, ±4g, ±8g, ±16g
Czulość pomiaru przyspieszenia	±2g ±4g ±8g ±16g	0.06 mg/LSB 0.12 mg/LSB 0.24 mg/LSB 0.49 mg/LSB
Początkowa, kalibrowana tolerancja od przyspieszenia ziemskiego		±50 mg
Pasma pomiarowe przyspieszenia (max.)		1 kHz

Tab. 3. Wybrane parametry ukladu AK8975

Tab. 3. Selected parameters of AK8975

Parametr	Warunki	Wartosc
Czulość pomiaru pola magnetycznego	25 °C	0,3 μT
Zakres pomiaru pola magnetycznego		± 1200 μT
Pasma pomiarowe		10 Hz
Rozdzielczosc pomiarowa		13 bitów

Istotnym zadaniem projektowania zadajnika jest określenie wzajemnego ułożenia obu ukladów pomiarowych na plytce modulu. Wynika to z różnej polaryzacji osi lokalnych ukladów odniesienia dla podzespolów. Uklad MPU-6000 mierzy parametry ruchu, tak jak pokazano na rys. 7a. Znakiem orientujacym jest kropka naniesiona na jego górnej powierzchni. Nicco inaczej wyglada orientacja ukladu odniesienia dla podzespolu AK8975 (rys. 7b).



Rys. 7. Orientacja i polaryzacja sygnalów dla ukladu MPU-6000 (a) i AK8975 (b)

Fig. 7. Orientation and polarization signals of MPU-6000 (a) and AK8975 (b)

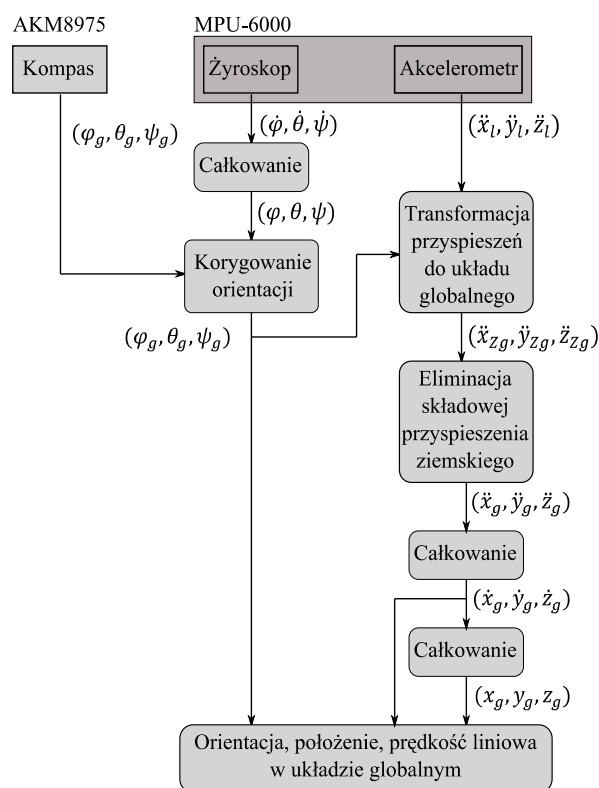
Aby zapewnić zgodność układów odniesienia obu komponentów elektronicznych (zwroty osi i ich kierunki), należy AK8975 obrócić o 90° względem osi Z, a następnie o 180° względem nowej osi X. Odpowiada to przyłutowaniu AK8975 od spodniej strony płytki drukowanej (jeżeli MPU-6000 zamontowano na wierzchniej). Nie jest to kłopotliwe, ponieważ oba układy mają grubość 0,9 mm i są przeznaczone do dwustronnego montażu.

Układ elektroniczny zaprojektowano zgodnie z dokumentami referencyjnymi producentów [2, 3]. Oprócz sensorów, płytkę zawiera kilkanaście biernych elementów (rezystory, kondensatory), stabilizator napięcia zasilania oraz miniaturowe złącze interfejsu do I²C i zasilania. Sterownik lokalny (rys. 6) to zewnętrzna płytkę prototypowa z układem LPC1796. Programowanie układów oraz wymiana danych z LPC1796 jest nieskomplikowana. Przykładowo, układ AK8975 z punktu widzenia programisty jest zbiorem dziewiętnastu 8-bitowych rejestrów, które w kilku przypadkach są łączone w pary, tak aby przechowywać 16-bitowe dane. Na sześciu rejestrach można wykonywać operacje zapisu/odczytu, pozostałe można tylko odczytywać. Opis rejestrów oraz ich funkcje w postaci komentarzy do kodu przedstawiono poniżej

```
// Rejestry układu AK8975
// rejestr identyfikacji urządzenia, domyślna wartość 0x48
#define AK8975_REG_WIA 0x00
// informacja o wersji urządzenia, pole firmowe
#define AK8975_REG_INFO 0x01
// rejestr statusowy nr 1
#define AK8975_REG_ST1 0x02
// dane osi X,y,z, bajt młodszy | starszy
#define AK8975_REG_HXL 0x03
#define AK8975_REG_HXH 0x04
#define AK8975_REG_HYL 0x05
#define AK8975_REG_HYH 0x06
#define AK8975_REG_HZL 0x07
#define AK8975_REG_HZH 0x08
// rejestr statusowy nr 2
#define AK8975_REG_ST2 0x09
// rejestr sterujący, ustawianie trybu pracy,...
#define AK8975_REG_CNTL 0x0A
// rejestr rezerwowany
#define AK8975_REG_RSVC 0x0B
// Rejestry układu AK8975 c.d.
// generowanie pola mag. -> funkcja autotestowania
#define AK8975_REG_ASTC 0x0C
// rejestr testowy: pole firmowe
#define AK8975_REG_TS1 0x0D
// rejestr testowy: pole firmowe
#define AK8975_REG_TS2 0x0E
// rejestr sterujący interfejsu I2C: domyślnie
// interfejs I2C włączony
#define AK8975_REG_I2CDIS 0x0F
// rejestr regulacji czułości w osi X, Y, Z
#define AK8975_REG_ASAX 0x10
#define AK8975_REG_ASAY 0x11
#define AK8975_REG_ASAZ 0x12
```

Współpraca z układem AK8975 rozpoczyna się od jego zainicjowania, czyli wpisania do rejestrów sterującego AK8975_REG_CNTL i statusowych AK8975_REG_ST1, AK8975_REG_ST2 wartości, które: wybierają tryb pracy, ustalają czułość, uruchamiają funkcję autotestowania i zerują układ. Po inicjalizacji układu, współpraca z nim ogranicza się do wywołania funkcji odczytu i zapisu danych, czyli: $ak8975_data_RD(.)$ i $ak8975_data_WR(.)$. Obie funkcje obsługują strukturę danych interfejsu I²C.

Układy MPU-6000, AK8975 generują pięć sygnałów, oznaczonych symbolicznie: $(\dot{\phi}, \dot{\theta}, \dot{\psi})$, $(\ddot{x}_l, \ddot{y}_l, \ddot{z}_l)$ oraz $(\phi_g, \theta_g, \psi_g) \rightarrow$. Pierwsze dwie trójki odczytywane są z układu MPU-6000 zaś ostatnia trójka z kompasu AK8975. Przyjęte nazwy kątów są zgodne z nomenklaturą stosowaną do opisu kątów Eulera w reprezentacji ZYX. Sekwencję operacji wykonywanych na powyższych trójkach danych przedstawiono graficznie na rys. 8.



Rys. 8. Schemat przetwarzania danych

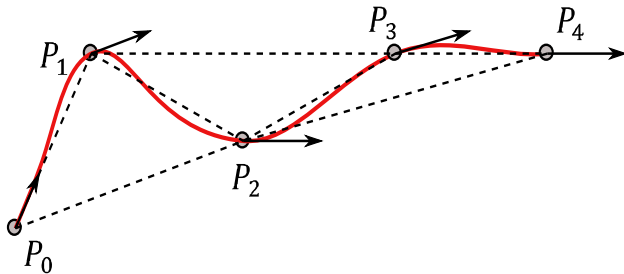
Fig. 8. Scheme of data processing

Na rys. 8 widoczne są trzy tory przetwarzania danych. Pierwszy z nich, pochodzący z kompasu wektor $(\phi_g, \theta_g, \psi_g)$ jest aktualizowany 10 razy na sekundę, czyli dość wolno. Pełni on rolę pomocniczą i służy do korygowania globalnej orientacji układu, mierzonej żyroskopem. Zdarza się, że dryfty termiczne lub inne zakłócenia działające na żyroskop mogą wpływać na rozkalibrowanie układu, w wyniku czego następuje kumulowanie błędów operacji całkowania. Błąd w bieżącej orientacji wyznaczany jest na podstawie wskazań kompasu i gdy pomiar żyroskopowy przekroczy dopuszczalny zakres, orientacja jest korygowana. Pomiar orientacji żyroskopowej odbywa się co 50 ms. Dzięki tak szybkiemu cyklowi obliczeń, błąd obliczeń numerycznych jest znikomy, nawet w długim okresie czasu. Wyznaczona

orientacja jest wykorzystywana do transformacji wektora przyspieszenia z układu lokalnego do globalnego, $(\ddot{x}_i, \ddot{y}_i, \ddot{z}_i) \rightarrow (\ddot{x}_g, \ddot{y}_g, \ddot{z}_g)$, za pomocą operatora rotacji $R_{RPY} = R_z(\phi_g)R_y(\theta_g)R_x(\psi_g)$. W następnym kroku od składowej \ddot{z}_g jest odejmowana wartość odpowiadająca przyspieszeniu ziemskiemu 1g. Po tym etapie wektor przyspieszeń liniowych jest całkowany kolejno dwa razy w celu wyznaczenia liniowej prędkości i globalnego położenia. Aktualizacja globalnego położenia odbywa się z cyklem 1 ms. Jest to szybkość wystarczająca do wytworzenia precyzyjnego manewru w przestrzeni. Przykładowo, dla szybkiego ruchu ręką z prędkością liniową 1 m/s, położenie zostanie określone z dokładnością około 1 mm.

4. Generowanie zadanej trajektorii w przestrzeni 3D

Punkty przecięcia przekątnych kolejnych czworokątów wypukłych, wyznaczonych na podstawie zdjęć CT, tworzą wektor punktów kontrolnych kubicznych wielomianów sklepanych dla reprezentacji *Catmull-Rom* [7]. Reprezentację tę cechuje bardzo dobre dopasowanie do interpretacji punktów kontrolnych otrzymanej krzywej. Ponadto, ma ona wbudowany mechanizm generowania wektorów prędkości w tych punktach. Zgodnie z opisem *Catmull-Rom*, krzywa kubiczna przechodzi przez wszystkie punkty kontrolne. Kierunek krzywej (styczna) w danym punkcie kontrolnym jest wyznaczany na podstawie sąsiednich punktów kontrolnych, jako ich różnica, czyli np. dla punktu p_i jest to $0,5(p_{i+1} - p_{i-1})$. Interpretację graficzną pokazano na rys. 9. Wektory styczne w punktach kontrolnych są równoległe do odcinków łączących pary (p_{i+1}, p_{i-1}) .



Rys. 9. Krzywa Catmull-Rom jako trajektoria ruchu robota ROCH-1
Fig. 9. Catmull-Rom curve as the trajectory of the robot motion

Segment takiej krzywej kubicznej jest określony wzorem :

$$p(u) = [1, u, u^2, u^3] \begin{bmatrix} 0 & 1 & 0 & 0 \\ -0.5 & 0 & 0.5 & 0 \\ 1 & -2.5 & 1 & -0.5 \\ -0.5 & 1.5 & -1.5 & 0.5 \end{bmatrix} \begin{bmatrix} p_{i-2} \\ p_{i-1} \\ p_i \\ p_{i+1} \end{bmatrix} \quad (1)$$

lub w postaci wielomianowej (2):

$$p(u) = p_{i-1} + (0.5p_{i-2} + 0.5p_i)u + (p_{i-2} - 2.5p_{i-1} + 2p_i - 0.5p_{i+1})u^2 + (-0.5p_{i-2} + 1.5p_{i-1} - 1.5p_i + 0.5p_{i+1})u^3 \quad (2)$$

Punkty kontrolne p_i krzywej w przestrzeni trójwymiarowej mają składowe X,Y,Z, czyli $p_i = p_{x,i}, p_{y,i}, p_{z,i}$. Możemy założyć, że podczas ruchu po krzywej w kierunku pola operacyjnego, orientacja narzędzia jest zgodna z wektorami T, N, B trójsianu Freneta, czyli:

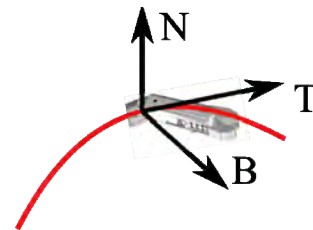
$$\begin{cases} T(u) = \frac{p'(u)}{\|p'(u)\|} \\ N(u) = \frac{T'(u)}{\|T'(u)\|} = \frac{p'(u) \times (p''(u) \times p'(u))}{\|p'(u)\| \|p''(u) \times p'(u)\|} \\ B(u) = T(u) \times N(u) \end{cases} \quad (3)$$

gdzie pochodne krzywej *Catmull-Rom* są określone wzorem:

$$p'(u) = (0.5p_{i-2} + 0.5p_i) + (2p_{i-2} - 5p_{i-1} + 4p_i - p_{i+1})u + (-1.5p_{i-2} + 4.5p_{i-1} - 4.5p_i + 1.5p_{i+1})u^2 \quad (4a)$$

$$p''(u) = (2p_{i-2} - 5p_{i-1} + 4p_i - p_{i+1}) + (-3p_{i-2} + 9p_{i-1} - 9p_i + 3p_{i+1})u \quad (4b)$$

Orientacja efektora, wyznaczana automatycznie wektorami T, N, B , (rys. 10), może być w dowolnej chwili zmieniona przez chirurga-operatora.



Rys. 10. Orientacja efektora za pomocą trójsianu Reneta
Fig. 10. Effector orientation with Frenet trihedron

Połączenie półautomatycznej i ręcznie generowanej trajektorii ruchu ma widoczne zalety. W pierwszej fazie znacząco odciąża chirurga-operatora, a po wykonanym zabiegu pozwala „odwrócić” kolejność ruchów w sposób automatyczny (tzw. odwrócenie mapy).

Kolejną funkcją, realizowaną programowo, która zwiększa jakość zadajnika trajektorii jest „eliminacja efektu drżenia rąk”. Zauważmy, że składowe X, Y, Z trajektorii ruchu są funkcjami czasu i mogą być poddawane filtracji. Drżenie rąk (np. pojawiające się w efekcie zmęczenia) ma częstotliwość rzędu 10–12 Hz. Jeżeli zastosujemy filtry dolnoprzepustowe, np. drugiego rzędu, z częstotliwością odcięcia o wartości około 8 Hz, efekt ten nie przeniesie się na drżenie robota.

Programowe przetwarzanie danych pozwala na łatwą archiwizację wykonywanych czynności podczas operacji. Umożliwia późniejszą ocenę manewrów, ich wizualizację i wskazuje na miejsca wystąpienia błędów oraz kierunki dalszej optymalizacji zabiegu.

5. Podsumowanie

Prezentowany sposób generowania trajektorii ruchu robota chirurgicznego za pomocą półprzewodnikowego sensora ruchu o 6 stopniach swobody jest fragmentem badań prowadzonych w ramach realizacji rozpoczętego w 2010 r. projektu finansowanego przez MNiSW. Generowana trajektoria obejmuje trzy fazy. Pierwsza faza polega na pół-automatycznym dojściu do pola operacyjnego. Faza druga to etap, w którym trajektoria wyznaczana jest na podstawie ruchu ręki operatora z przytwierdzonym sensory typu MEMS. Chirurg-operator wykonuje manewry, które są skalowane, dzięki czemu uzyskuje się większą precyzję. Dodatkowo, w torze przetwarzania parametrów ruchu, włączone są filtry eliminujące efekt drżenia rąk. Jest to zabieg poprawiający pozycjonowanie w statycznych fazach operacji.

Istotnym elementem projektu zadajnika było zastosowanie takich rozwiązań, których funkcjonalność byłaby realizowana głównie programowo. Dzięki proponowanemu podejściu, urządzenie może być doskonalone etapami. Zdaniem autora, zaletą opisanego powyżej rozwiązania jest również niska cena oraz możliwość wykorzystania go w warunkach polowych (mobilne ambulatoria wojskowe).

Praca realizowana w ramach projektu MNiSW N514 237638

5.1. Bibliografia

1. Podseńkowski L.: *Roboty medyczne – budowa i zastosowanie*, WNT, 2010.
2. *MPU-6000 and MPU-6050 Product Specification*, Revision 3.2., InvenSense Inc. USA, 2011.
3. *AK8975/AK8975B 3-axis Electronic Compass*, Asahi KASEI, 2009.
4. Leniowski R., Pajda R., Leniowska L., Cieślak J.: *Projekt wieloczołowego manipulatora chirurgicznego nowej generacji*, Politechnika Warszawska, Prace naukowe Elektrotechnika z 175, „Problemy Robotyki”, T. 1, Warszawa 2010, 63–78.
5. *Digital Imaging and Communications In Medicine (DICOM), Part3: Information Object Definition*, National Electrical Manufacturers Association, USA 2009.
6. Leniowski R.: *Deterministic and Heuristic solution of Kinematic Problems of surgical robot ROCH-1*, “Archives of Mechanics”, (to be published).
7. Catmull E., Rom R.: *A class of local interpolating splines*. “Computer Aided Geometric Design”, 1974, 317–326. ■

Generating spatial trajectories for surgical robot using a 6 DOF semiconductors motion sensors

Abstract: The paper presents an alternative approach to the construction of the surgical robot trajectory haptic console by the use of motion MEMS type sensors: compass, gyro, accelerometer. Miniature module with these elements can be applied directly to the surgeon's hand or can be located on his fingers. The advantage of this device is not only low price, but also a high level of functionality associated with openness to adding new software modules. An example of built-in function can be the generator of a come path to the surgery field by the use of spline polynomials in the Catmull-Rom representation. In addition, the haptic console is very small and easy to use in mobile systems (military field ambulances).

Keywords: robotics, medical robots, trajectory planning, motion sensors

dr inż. Ryszard Leniowski

Absolwent Politechniki Rzeszowskiej w specjalności automatyka. Stopień doktora nauk technicznych (z wyróżnieniem), w zakresie automatyki i robotyki, uzyskał na Wydziale Automatyki, Elektroniki i Informatyki, Politechniki Śląskiej w Gliwicach. Pracuje w Katedrze Informatyki i Automatyki Politechniki Rzeszowskiej na stanowisku adiunkta. W latach 1995–97 przebywał na stypendium naukowym na Uniwersytecie Technicznym w Lyngby, Dania. Realizował kilka projektów krajowych jako kierownik i główny wykonawca. Opublikował ponad 50 prac naukowych. Jego zainteresowania badawcze dotyczą lekkich robotów, miniaturowych robotów chirurgicznych, modelowania matematycznego, z wykorzystaniem algebry komputerowej, algorytmów sterowania ruchem oraz zastosowań grafiki komputerowej do wizualizacji procesów dynamicznych.



e-mail: lery@prz-rzeszow.pl