

조선용 3차원 그래픽 해석 시뮬레이션 시스템^{*1)}

전 희 성
컴퓨터공학과

〈요 약〉

조선업계에서는 가공, 소조립, 지상조립, 선대조립등의 선박 제작 과정 중 선각(블럭)이 설계치에 맞게 제작되고 있는 지 확인하는 과정에 고정밀도의 3차원 측정 시스템을 사용하고 있다. 본 논문에서는 이 측정 시스템의 활용도를 높이는 “조선용 3차원 그래픽 해석 시뮬레이션 시스템”을 개발하였다.

본 시스템은 3차원 측정 시스템으로부터 전송되는 측정 데이터를 PC의 파일로 저장할 수 있는 DOS용 프로그램과 전송된 측정 데이터및 이에 대응되는 설계 데이터를 이용하여 제작된 블럭의 3차원 데이터 해석및 시뮬레이션을 할 수 있는 한글 윈도우용 프로그램으로 구성되어 있다.

다양한 설계 데이터와 이에 대응되는 측정 데이터를 구할 수 없었던 관계로 이 두 데이터를 모의로 만들어 시뮬레이션을 수행하여 그 결과를 포함하였다.

Three-Dimensional Graphic Analysis and Simulation System for Shipbuilding

Hee sung Jun
Department of Computer Engineering

〈Abstract〉

It is important to control the accuracy of fabrication, assembly, and shipwright in shipbuilding industry. An accurate measurement system has been used for large-scale ship construction. With the help of this measurement system, we developed a

1) * 본 연구는 현대중공업(주) - 울산대학교의 협동과제로 수행되었음.

three-dimensional graphic analysis and simulation system to evaluate ship block assembly and shipwright.

The system is composed of a DOS program which receives and saves measured data into a file and Windows program which displays the deviation between the design block shape and the measured block shape on a PC screen.

We did the simulation to test the efficiency of the developed system with pseudo design and measured data set.

1. 서 론

현재 국내 조선업계에서는 AutoDEF 또는 StearBear등의 조선 전용 CAD 시스템을 사용하여 선박의 선체부를 설계하고 있으며[3,4], 선박의 제작은 이 설계치에 따라 가공, 조립, 지상조립, 선대 조립등의 여러 단계를 거쳐 이루어지고 있다[5]. 이 때 각 단계별로 설계치에 맞게 제작되고 있는 지 확인할 필요가 있으며 만약 허용 오차보다 큰 오차가 발생했을 경우, 다음 단계의 작업에 들어 가기 전에 수정 작업이 요구된다.

현대중공업의 경우 이 확인 과정에 일본의 SOKKIA사가 Optomechatronics 기술을 사용하여 개발한 고정밀도 3차원 측정 시스템인 MONMOS 시스템(Mono, Mobile, 3-D Station의 약칭)을 사용하고 있으며 이 시스템은 다음과 같은 세 부분으로 구성되어 있다.

- 대상물을 관측하는 계측기인 3-D Station NET2
- 좌표치를 산출하는 SDR4C 제어 단말기(control terminal)
- 측정하려고 하는 점의 목표로 사용하는 반사판

그림 1에 반사판을 제외한 MONMOS 시스템이 도시되어 있다.

현재는 이 장비를 이용하여 제작된 블록의 주요 지점들의 좌표를 측정한 후 계산기를 이용하여 제작 오차를 계산하고 있다. 그러나 이 방법은 수작업으로 오차를 계산하는 과정 중 작업자가 실수 등으로 계산을 틀리게 할 우려가 있으며, 계산 시간 또한 오래 걸리는 형편이다. 더구나 실제로 제작된 블록의 입체적인 형상을 파악하기도 쉽지 않은 일이다.

SDR4C 제어 단말기의 기능 중 하나인 직렬 단자를 이용한 화일 전송 기능을 사용하면, 이 제어 단말기를 PC의 RS-232C 직렬 단자와 연결하여, 실측한 선박의 데이터를 PC로 전송할 수 있다. 만약 측정된 데이터 화일을 PC로 전송하고, 전송된 데이터 화일들을 이용하여 시뮬레이션을 할 수 있다면 선박 제작 과정 중 측정 및 확인 과정에서 발생하는 시간의 손실을 줄일 수 있을 것이다.

이러한 목적으로 일본의 Ishikawajima사에서 개발된 프로그램인 3D Analysis System[6]이 있으나 가격이 매우 비싸며 한국의 실정에도 맞지 않는다. 또한 이 프로그램을 구입하여 데이터 화일을 전송받을 수 있게 된다 하더라도 현대중공업에서 선박 설계에 이용하고 있는 AutoDEF CAD 시스템에서 사용하는 데이터와 호환이 되도록 하는 (즉 설계 데이터와 측정 데이터의 포맷을 맞게 변환을 하는) 노력이 추가로 필요하다.

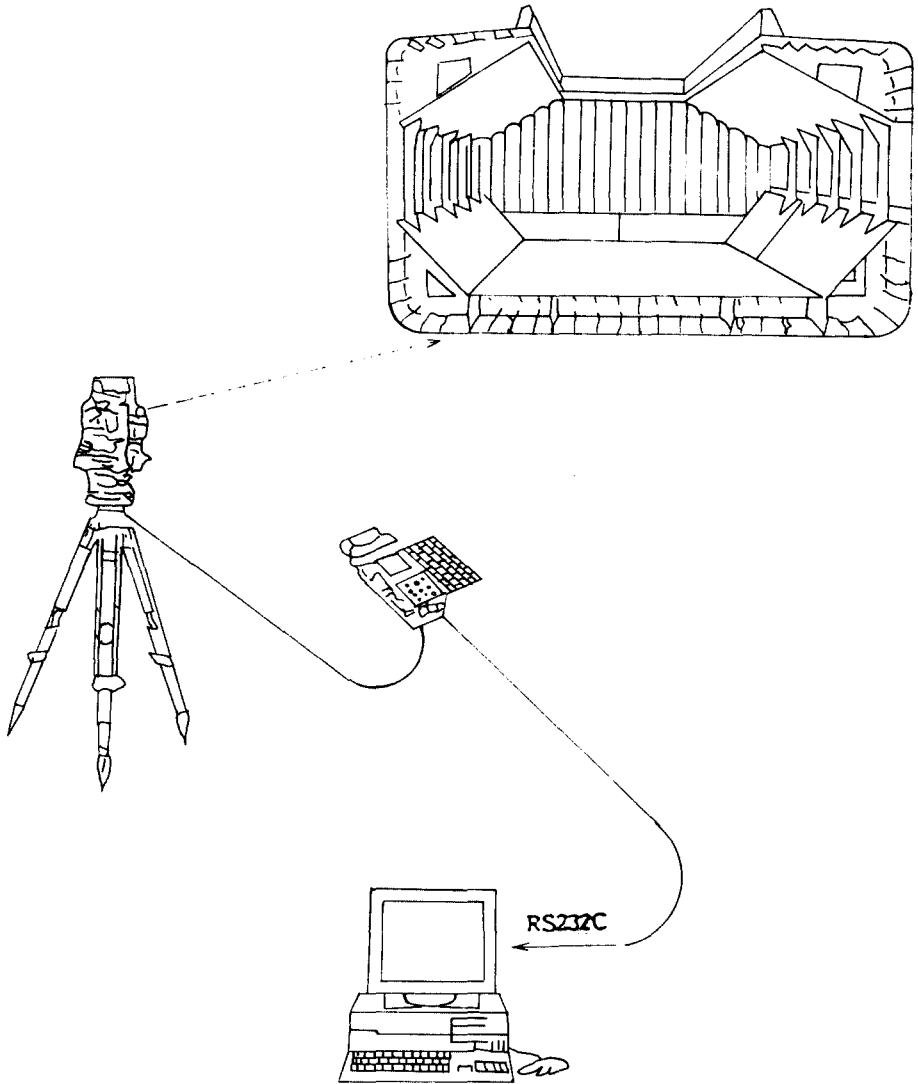


그림 1. MONMOS 시스템의 구성도

이와 같은 필요에 따라 본 연구의 주요 목표는 이러한 두 형태의 데이터 비교 분석을 좀 더 편하고 정확하게 하기 위해 PC에서 운용되는 “3차원 그래픽 해석 시뮬레이션 시스템”을 개발하는 것이다.

2. 시스템의 구성

본 시스템은 크게 다음의 3개 부분으로 구성되며

- 측정 데이터 관련부
- 설계 데이터 관련부
- 3차원 그래픽 해석 및 시뮬레이션 프로그램

본 시스템을 구성하는 각 부분에 대해 살펴보면 다음과 같다.

2.1 측정 데이터 관련 부분

측정 좌표계는 그림 2에서와 같이 임의의 두 점을 관측하는 것으로 설정이 가능한 데 첫번째 계측점이 자동적으로 좌표계의 원점(origin)이 되며, 두번째로 계측한 점이 xz -평면상의 점 $(X_a, 0, Z_a)$ 가 된다. 이 두 점의 설정은 임의이기 때문에 계측물 위의 점이거나 외부점이라도 상관 없다. 그림 2에서 보듯이 측정 좌표계는 우수 좌표계(right-hand coordinate system)이다.

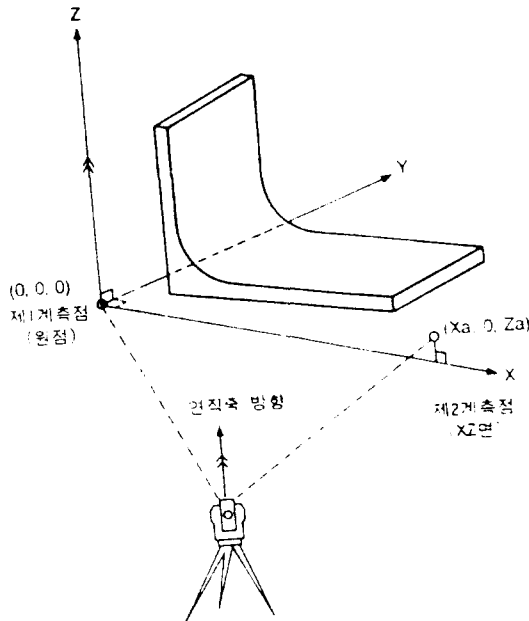


그림 2. 측정 좌표계의 설정

이와 같이 측정좌표계를 설정한 후에 각 계측점의 좌표는 그림3에서와 같이 각 계측점의 거리, 수평각, 연직각의 세 요소가 측정되고 좌표는 이에 따라 자동으로 계산된다.

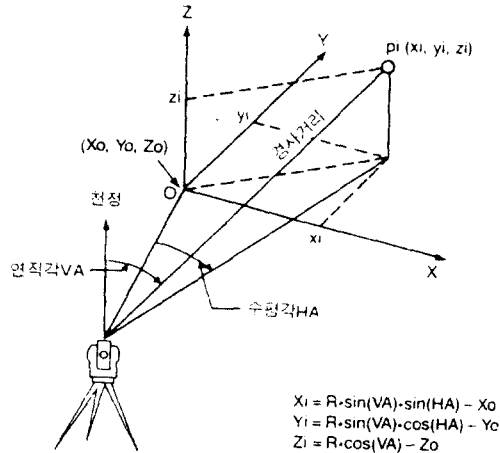


그림 3. 측정점의 좌표계산

이와 같이 계측되어 제어 단말기 SDR4C에 기억되어 있는 측정 데이터를 직렬 인터페이스를 통해 PC로 전송하여 파일로 저장하는 프로그램의 개발이 이 부분의 주요 내용이다.

2.2 설계 데이터 관련 부분

선박의 설계에 사용하는 좌표계도 그림 4와 같이 설정된 우수 좌표계이다. 그러나 선박의 설계에 이 하나의 좌표계만 사용되는 것이 아니다. 선박 하나를 제작하는 데 필요한 블록의 수는 평균 200 - 400개 정도이므로, 통상 선박 전체를 표시하는 하나의 전체 좌표계(global coordinate)를 사용하기 보다는 각 블록의 국부 좌표계(local coordinate)를 사용하여 선박의 설계 데이터를 표시한다. 이때 각 블록의 상호 관계를 알아야 할 필요가 있으므로 각 블록의 국부 좌표계로 표시된 좌표값은 필요시 선박의 전체 좌표계로 변환이 가능해야 한다.

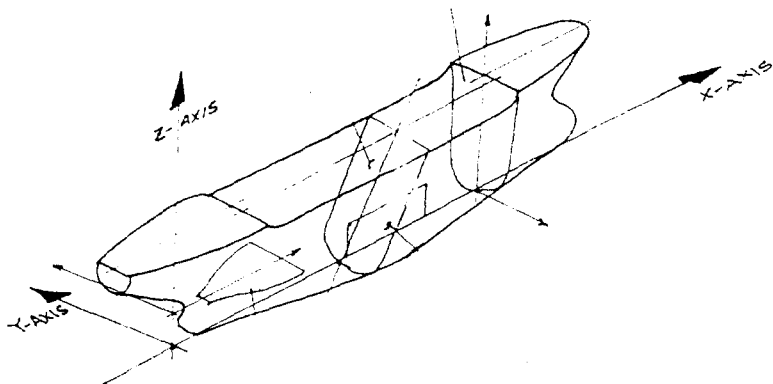


그림 4. 설계 좌표계

2.3 3차원 데이터 해석 및 시뮬레이션부분

3차원 데이터 해석 및 시뮬레이션 프로그램은 두 종류의 데이터(측정 및 설계 데이터)를 이용하여 블록 데이터의 해석에 필요한 여러 기능들을 수행하고 측정 결과와 해석의 결과를 화면에 표시하고 프린터로 인쇄도 가능하도록 한다.

3. 해석 시뮬레이션의 주요 기능 설계

3.1 측정치와 설계치의 비교

선박의 블록이 설계에 맞게 정확히 제작되었는지 확인하려면 먼저 좌표계의 변환이 필요하다. 이 경우 측정 데이터를 설계 데이터와 비교해야 하므로 측정 데이터 좌표계로 정의된 측정 데이터를 설계 좌표계로 변환해야 한다. 이 변환 방법에는 자동과 수동의 두 가지 방법이 있다.

3.1.1 자동 좌표 변환 방법

먼저 측정 좌표계로부터 설계 좌표계로의 변환을 자동으로 해 주는 방법에 대해 알아보자. 3차원 공간에서 점 $P(x, y, z)$ 가 회전 및 이동을 하여 점 (P', x', y', z') 으로 변환이 되었을 경우, 이것을 rigid body motion이라 하고 이를 식으로 쓰면 다음과 같다[12].

$$P' = R P + t \quad (1)$$

이 식에서 R 과 t 는 각각 회전 행렬과 이동벡터이다.

식 (1)이 rigid motion의 올바른 모델이 되기 위한 필요조건은 R 행렬이 orthonormal이어야 하고 충분조건은 $\det(R) = 1$ 이어야 한다[11].

우리들이 사용하는 설계 데이터와 측정 데이터를 이용하면 대응되는 3차원 데이터로부터 motion parameter를 추정할 수 있는데, 각 점마다 대응되는 3차원 데이터 쌍이 있으므로 남아 있는 문제는 다음과 같이 기술할 수 있다.

문제: 주어진 두 데이터 셋 $\{p_i\}$, $\{q_i\}$ ($i = 1, 2, \dots, n$)에 대해 최소 평균 자승 오차 $e^2(R, t, s)$ 를 갖도록 하는 유사 변환(similarity transformation) 파라미터 (회전: R , 이동: t , 배율: s)를 구하라. 단

$$e^2(R, t, s) = \frac{1}{n} \sum_{i=1}^n \| q_i - (sR p_i + t) \|^2 \quad (2)$$

본 연구에서는 위의 식을 풀기 위해 Umeyama에 의해 개발된 방법[13]을 사용하였다. 3차원 공간에서 일직선 상에 있지 않는 3개 이상의 점이 있으면 변환 매개변수들은 유일하게 결정된다는 것이 [13]에 제시되어 있으며 그 정리를 3차원의 경우에 대해서는 다음과 같이 요약할 수 있다.

$P = \{p_1, p_2, \dots, p_n\}$ 와 $Q = \{q_1, q_2, \dots, q_n\}$ 를 3차원 공간에서의 대응되는 위치 벡터(점 데이터), m_p 와 m_q 를 각각 P와 Q의 평균 벡터, σ_p^2 를 P의 분산 벡터(variance vector), σ_{pq} 를 P와 Q의 분산 행렬(covariance matrix)이라 하고, 만약 σ_{pq} 의 singular value decomposition이 UDV^T ($D = \text{diag}(d_1), d_1 \geq d_2 \geq d_3 \geq 0$ 와 같으면, $\text{rank}(\sigma_{pq}) \geq 2$ 일때 motion 매개변수는 다음과 같이 유일하게 결정된다.

$$R = USV^T$$

$$t = m_q - sRm_p \tag{3}$$

$$s = \frac{1}{\sigma_p^2} \text{trace}(DS)$$

단

$$S = \begin{cases} I & \text{if } \det(U)\det(V) = 1 \\ \text{diag}(1, 1, -1) & \text{if } \det(U)\det(V) = -1 \end{cases}$$

자세한 정리와 증명은 참고문헌 [13] 을 참고하기 바람. 위의 식 (3)에서 R과 t를 구하면 측정 데이터는 설계 데이터의 좌표계로 쉽게 변환된다.

3.1.2 수동 좌표 변환 방법

이 방법은 사용자가 화면상에서 대화식으로 좌표변환을 할 수 있는 방법이다.

예를 들어 블록의 일 부분인 형상 cab와 형상 c'a'b'가 그림 5와 같이 놓여 있다 가정 하자. 이 경우 편의상 형상 cab를 설계치, c'a'b'을 측정치라 하고, 제작 또는 계측의 오차로 인해 $ca \neq c'a'$, $ba \neq b'a'$ 이고 $\angle cab \neq \angle c'a'b'$ 라 하자. 이 경우 우리가 할 수 있는 최선의 좌표 변환은

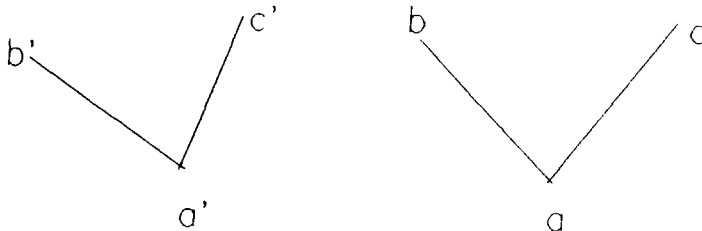


그림 5. 수동 좌표 변환

1. 한 점은 완전히 일치시키고
2. 그 다음 한 선분의 방향을 일치시킨 후
3. 마지막으로 평면을 같은 방향으로 놓이게 하는 것이다.

예를 들어 설명하면

1. 먼저 점 a' 를 점 a 와 일치시킨다. 이 경우의 변환은 단순한 3차원 이동 변환이다. 이 변환에 의해 점 b' 과 c' 도 변환이 되어 점 b'' 과 c'' 이 될 것이다.
2. 그 다음 점 a 를 축으로 하여 회전시킴. 회전축은 벡터 ab 와 ab'' 의 외적으로부터, 회전 각도는 두 벡터의 내적으로부터 구할 수 있다. 이 변환에 의해 점 b'' 과 c'' 이 다시 점 b''' 과 c''' 으로 변환될 것이다.
3. 그 다음 면 cab 와 면 $c'''a'''b'''$ 을 같은 평면 상에 이케 하려면 축 ab 를 중심으로 어떤 각도만큼 회전하면 되는 데, 이 회전 각도는 이 두 평면의 단위 법선 벡터로부터 구할 수 있다.

위에서 단계 2, 3의 회전 변환은 임의의 축을 중심으로 하는 회전 변환으로 3차원 그래픽에서 임의의 축을 중심으로 하는 회전 변환은 3차원 이동 및 x , y , z 축 중심의 회전에로 구현할 수 있다.

즉 회전의 중심 축이 벡터

$$V = (x_2 - x_1, y_2 - y_1, z_2 - z_1)$$

으로 주어지고 V 의 단위 벡터를 $u = (a, b, c)$ 라 할때 회전 각도가 θ 일 경우의 회전 행렬은

$$R(\theta) = TR_x(a)R_y(\beta)R_z(\theta)R_y^{-1}(\beta)R_x^{-1}(a)T^{-1}$$

와 같이 계산되며 여기서

$$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -x_1 & -y_1 & -z_1 & 1 \end{bmatrix}$$

이고 회전 행렬 R_x , R_y , R_z 는 각각 x , y , z 축을 중심으로 하는 회전을 의미한다. 자세한 내용은 참고 문헌 [9, 10]을 참조바람.

4. 구현

본 시스템의 구현은 앞에서 기술한 대로 세 부분의 일로 나누어 진행하였으며, 각 프로그램 모듈들을 통합하여 하나의 큰 프로그램으로 만들 수도 있으나, 개발, 운용 및 유지 보수 편의성을 고려하여 별개의 프로그램으로 구성하였다.

4.1 측정 데이터 관련

SDR4C의 사용자 설명서 [1]에 기술된 데이터의 전송 규약에 따라 RS-232C 직렬 인터페이스를 이용한 측정 데이터의 수신 프로그램을 작성하였다. 표1에 데이터의 전송 규약이 요약되어 있다.

표 1. 데이터의 전송규약

전송속도	1200 baud
데이터길이	8 bits
stop bit	1 bit
parity bit	없음
X parameter	유효

측정 데이터의 수신 프로그램은 직렬 단자의 사용이 가능한 286이상의 PC에서 사용이 가능하도록 MS-DOS 환경에서 Turbo C 2.0 컴파일러를 사용하여 개발하였다. 이 프로그램은 참고도서 [2] 의 3장에 있는 pt2pt.c 프로그램을 응용하여 작성하였으니 자세한 내용은 [2]를 참고하기 바람.

그림6에 측정데이터의 포맷을 나타내었다.

```

(1) (1)(1)
0x02 CR LF <--Starting flag of sending data
(8) (1)(1)
File name CR LF <--File name
(8) (1) (8) (1) (8) (1)(1)
Material temp. SP Expansion coef. SP Scaling const. CR LF <--Parameters
(3) (1) (14) (1) (14) (1) (14) (1) (12) (1)(1)
No. SP Coordinate X SP Coordinate Y SP Coordinate Z SP Memo CR LF
.
.
.
No. SP Coordinate X SP Coordinate Y SP Coordinate Z SP Memo CR LF
(1) (1)(1)
0x03 CR LF <--Ending flag of sending data
    
```

Coordinate data

그림 6. 측정 데이터의 포맷

그림 6에서 괄호 안의 숫자는 바이트수를 나타내며 SP는 1 바이트의 blank space(16진수로 '20')를 뜻한다. 이 포맷은 각 측정점의 x, y, z 좌표값(coordinate data)뿐만 아니라 측정의 시작 및 종료 시각, 물체의 온도와 팽창 계수 등의 정보를 포함하고 있다. 이러한 정보들은 블록의 입체적인 형상을 표현하는 데는 필요없는 정보이므로 차후에 설명할 '3차원 그래픽 해석 시뮬레이션 프로그램'에서 사용하기 위해서는 파일의 각 행이 측정점의 3차원 좌표값만을 포함하도록 파일의 내용을 수정해야 한다.

현재까지는 제작된 블록의 주요 점의 좌표값을 측정할 때 그림 6에 표시된 12 바이트의 memo field를 거의 사용하지 않았다. 그러나 측정 데이터 파일을 이용하여 제작된 블록의 입체적인 형상을 PC 화면에서 알 수 있으려면, 블록의 각 점들의 3차원 좌표를 측정할 때, memo field에 각 점들과 서로 연결되는 연결점들의 고유 번호를 표시해 주어야 한다.

예를 들어 어떤 데이터 파일의 특정 행이

8 0 14000 4335 4-7-12-16

과 같다면, 8번점의 좌표값이 (0, 14000, 4335)이고 4, 7, 12, 16번점이 이 8번점과 연결되어 있다는 것을 의미한다. 이때 점과 점의 구분은 '-'로 하며 memo field의 맨 첫글자가 '.'이면 이 memo field의 내용은 주석(comment)임을 의미함.

4.2 설계 데이터 관련

실측치와 설계치의 비교를 위해서는 파일 포맷의 통일이 필요하다. 이 데이터 파일의 포맷은 실측치의 데이터 파일 포맷을 이용하기로 하는 데 이 포맷은 한 점의 3차원 데이터가 한 행씩 표시되는 ASCII 파일로 각 행의 자세한 포맷은 다음과 같다.

점	SP	x	SP	y	SP	z	SP	memo
(3)	(1)	(14)	(1)	(14)	(1)	(14)	(1)	(12)

- 괄호안의 숫자는 바이트 수를 나타내며 SP를 제외한 field들은 허용 최대치이며 SP는 위의 표에서는 1 바이트의 blank space를 뜻하나 이는 최소치임. 즉 연속되는 둘 이상의 blank space를 사용할 수 있음.
- label은 각 점의 고유 번호를 의미하므로 한 데이터 파일 안에서 서로 다른 점이, 같은 고유 번호를 사용할 수 없음.
- x, y, z 좌표값은 선박의 국부 또는 전체 좌표계로 표시된 실수값이며 단위는 mm를 사용함.
- memo field에는 해당점과 연결된 모든점들의 고유 번호를 표시하며 번호와 번호 사이는 '-'로 구분함.
- memo field에서 주석(comment)은 '.'으로 시작함.
- 각 행의 끝은 16진수 '0D'(Carriage return)과 '0A'(Line feed)로 끝남.

위의 사항에 추가하여 CAD 데이터 파일의 맨 첫줄에 선박의 전체 좌표와의 관계를 나타내야 한다. 예를 들어 파일의 맨 처음줄이

```
% 1000 2000 3000 0 90 0
```

과 같다면(이 행은 반드시 %로 시작해야함), 이 블록의 국부 좌표계의 원점이 선박의 전체 좌표계의 원점에서 y축을 중심으로 90도 회전 후, 각각 x, y, z 축 방향으로 각각 1000, 2000, 3000 mm 만큼 이동시킨 것과 같다.

4.3 3차원 데이터 해석 시뮬레이션부

3차원 데이터 해석 시뮬레이션 프로그램은 Microsoft의 한글 윈도우 3.1 환경에서 Borland C++ 3.1 컴파일러를 사용하여 개발하였다. 프로그램의 작성시 참고문헌 [7,8]을 주로 참고하였다. 이 프로그램의 개요 및 주요 기능에 대해 간단히 설명하면

- 본 프로그램은 한글 윈도우 3.1에서 수행된다.
- 사용자 인터페이스는 다른 윈도우용 프로그램과 유사한 표준 인터페이스를 따랐다.
- 하나의 블록 좌표 파일을 읽어들이어 블록을 화면상에서 표현 할 수 있다.

- 두개의 대응되는 블럭 좌표 화일(설계치및 측정치)을 읽어 들여 좌표의 비교를 통해서 설계 데이터에 대한 측정 블럭의 좌표/길이 값을 알 수 있고, 좌표값을 변경하여 변경된 좌표값에 대한 블럭형태를 비교 분석 할 수 있다.
 - 여러 개의 인접한 블럭 좌표 화일들을 읽어 들여 블럭의 이동에 의해서 블럭간의 결합을 화면 상에서 확인할 수도 있다.
 - 많이 쓰이는 기능들을 icon화한 도구 버튼(tool button)도 제공하므로, 이러한 버튼들을 누르면 메뉴에서 선택한 것과 동일한 효과를 얻을 수 있다.
- 그림 7에 본 시스템의 사용자 인터페이스를 나타내었다.

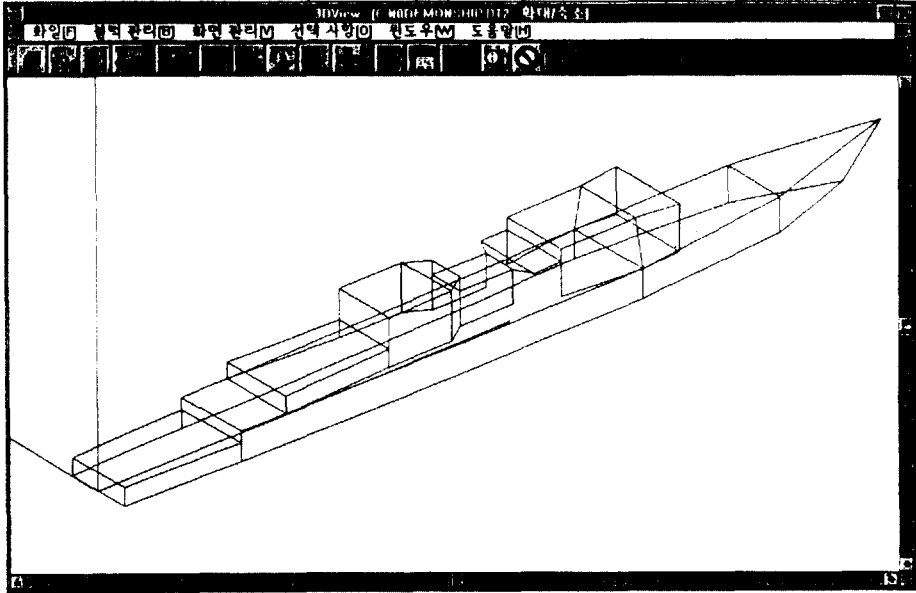


그림 7. 프로그램의 사용자 인터페이스

5. 시뮬레이션 및 검토

이상과 같이 개발한 시스템의 유용성을 확인하기 위해 시뮬레이션을 하였다. 시뮬레이션은 데이터의 오차가 전혀 없는 경우와 오차가 있는 경우로 나누어 실시하였다.

5.1 오차가 없는 경우

먼저 그림 8의 모양과 같은 블럭의 3차원 데이터가 표2에 기록되어 있다.

이 데이터를 3차원 공간에서 각각 x, y, z축을 중심으로 5, 10, 15도씩 회전시킨 후, x, y, z 축 방향으로 각각 50 100 150 만큼 이동시켜 만든 데이터의 내용은 표3과 같고 이 데이터를 실측 데이터로 이용하였다. 표3의 데이터는 표 작성의 편의상 소수점 세째 자리에서 반올림하였음.

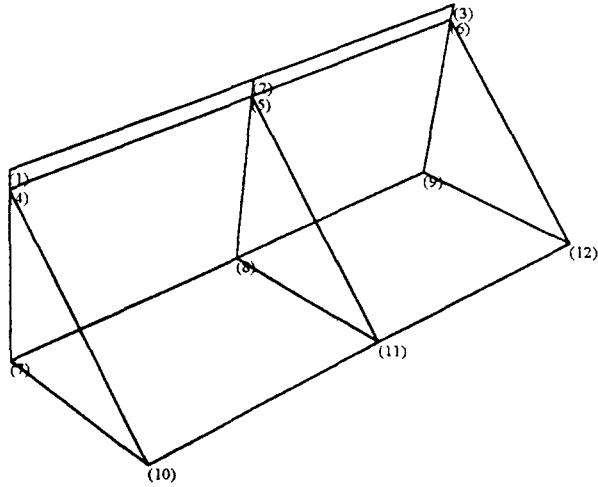


그림 8. Test 블록의 형상

이상의 설계 및 측정 데이터를 이용하여 먼저 자동 좌표 변환을 이용하여 변환 매개변수를 먼저 구하고 난 후 이에 따라 측정 데이터를 설계 데이터 좌표계로 변환을 하였다. 이 자동 변환 후의 데이터 파일의 내용은 표4와 같다.

그 다음 3.1.2절에서 기술한 방법을 이용하여 각 단계에서 점 1, 2, 7을 택하여 수동 변환을 한 결과가 표5에 기록되어 있다.

표2. Test블록의 설계 데이터

점	X	Y	Z	연결점
1	0	0	4735	2-4
2	0	7000	4735	1-3-5
3	0	14000	4735	2-6
4	0	0	4335	1-5-7-10
5	0	7000	4335	2-4-6-8-11
6	0	14000	4335	3-5-9-12
7	0	0	0	4-8-10
8	0	7000	0	5-7-9-11
9	0	14000	0	6-8-12
10	4435	0	0	4-7-11
11	4435	7000	0	5-10-10-12
12	4435	14000	0	6-9-11

표3. Test 블록의 측정 데이터

점	x	y	z
1	872.22	-306.41	4795.32
2	-911.98	6401.92	5698.02
3	-2696.18	13110.30	6600.73
4	802.76	-272.08	4402.90
5	-981.44	6436.26	5305.60
6	-2765.64	13144.60	6208.30
7	50.00	100.00	150.00
8	-1734.20	6808.33	1052.70
9	-3518.40	13516.70	1955.41
10	4268.80	1308.32	-491.01
11	2484.60	8016.66	411.69
12	700.40	14725.00	1314.39

표4. 자동 변환 후의 측정 데이터

x	y	z	오차
0.003734	-0.001163	4735.000000	0.003911
-0.000340	7000.000000	4735.000000	0.000340
0.007491	14000.000000	4735.000000	0.007491
0.003206	0.000070	4335.000000	0.003207
0.001040	7000.000000	4335.000000	0.001040
-0.001533	14000.000000	4335.000000	0.001533
0.003724	0.001133	0.002173	0.004512
-0.000622	7000.000000	0.001385	0.001519
0.004485	14000.000000	0.006975	0.008292
4435.000000	0.003646	0.001590	0.003978
4435.000000	7000.010000	0.002887	0.010408
4435.000000	14000.000000	0.000434	0.000434

표5. 수동 변환 후의 측정 데이터

x	y	z	오차
-0.000000	0.000000	4735.000000	0.000000
0.0000000	6999.995838	4735.000000	0.000000
0.011905	14000.039924	4735.005605	0.015548
-0.000527	0.001192	4335.004045	0.001303
0.001380	7000.003738	4335.003444	0.001380
0.002881	14000.008456	4335.002498	0.002881
0.000000	0.002010	0.001211	0.002347
-0.000272	6999.996890	0.001297	0.001325
0.008908	14000.031393	0.007760	0.011814
4434.997976	0.000013	0.001492	1.001492
4434.999994	7000.004863	0.003663	0.003663
4435.003542	14000.008943	0.002083	0.002083

표4와 5를 표2와 비교하면 두 데이터 모두 설계 데이터와 거의 근사한 것을 알 수 있으며, 자동및 수동변환 방법의 오차 평균은 각각 0.003889, 0.003653으로 별 차이가 없다. 따라서 블록의 제작및 측정 오차가 전혀 없는 경우에는 자동및 수동 변환의 두 방법 모두 다 사용가능한 것을 알 수 있다.

5.2 오차가 있는 경우

앞에서 다룬 오차가 없는 경우는 실제로는 거의 불가능한 경우라 생각할 수 있는 데, 실제의 경우 블록의 제작오차및 측정오차 때문에 앞에서 고려한 오차가 전혀 없는 경우는 기대하기 힘들다.

이러한 경우를 고려하여 앞에서 사용한 블록의 3차원 데이터를 약간 변형한 후(1번 점의 z값을 4740으로) 앞에서와 마찬가지로 3차원 공간에서 각각 x, y, z축을 중심으로 5, 10, 15도씩 회전시키고 또 x, y, z 축 방향으로 각각 50 100 150 만큼 이동시켜 만든 데이터의 내용은 표6와 같고 이 데이터를 측정 데이터로 이용하였다.

이상의 설계및 측정 데이터를 이용하여 먼저 자동 좌표 변환을 이용하여 변환 매개변수를 먼저 구하고 난 후 이에 따라 측정 데이터를 설계 데이터 좌표계로 변환을 하였다. 이 자동 변환 후의 데이터 화일의 내용은 표7과 같다.

수동 변환의 경우 앞에서와 유사하게 각 단계에서 점 7, 9, 3을 택하여 변환한 결과가 표8 에 기록되어 있다.

표7 의 자동변환의 오차 평균은 0.827702이며 수동변환의 오차 평균은 0.421630이다. 이 결과에서 보면 알 수 있듯이 오차가 있는 경우 수동 변환이 자동 변환보다 더 나은 변환 결과를 제시하며 또한 수동 변환의 경우 사용자가 변환하고자 하는 순서와 점을 선택해서 할 수 있으므로 더 다양한 시뮬레이션이 가능하다.

표6. 블럭의 측정 데이터(오차가 있는 경우)

점	x	y	z
1	873.09	-306.84	4800.23
2	-911.98	6401.92	5698.02
3	-2696.18	13110.30	6600.73
4	802.76	-272.08	4402.90
5	-981.44	6436.26	5305.60
6	-2765.64	13144.60	6208.30
7	50.00	100.00	150.00
8	-1734.20	6808.33	1052.70
9	-3518.40	13516.70	1955.41
10	4268.80	1308.32	-491.01
11	2484.60	806.66	411.69
12	700.40	14725.00	1314.39

표7. 자동변환 후의 측정 데이터(오차가 있는 경우)

x	y	z	거리(오차)
-0.137515	-0.008803	4739.05	4.052344
-0.099791	6999.99	4734.59	0.422088
-0.051288	1400.00	4735.14	0.149099
-0.115796	0.022925	4334.05	0.957306
-0.077291	7000.03	4334.59	0.418299
-0.039191	14000.00	4335.14	0.145382
0.113621	0.359316	-0.955389	1.027028
0.149947	7000.36	-0.409953	0.565814
0.195725	14000.40	0.141864	0.467369
4435.11	0.335593	-0.723791	0.805355
4435.15	7000.34	-0.176269	0.411304
4435.19	14000.30	0.367502	0.511036

표8. 수동변환 후의 측정 데이터(오차가 있는 경우)

x	y	z	거리(오차)
-0.003862	0.001021	4740.003343	5.000002
-0.007451	6.999.996043	4734.995514	0.007451
0.0000000	14000.040129	4734.997845	0.000000
-0.003271	0.001210	4335.002834	0.003488
-0.005817	7000.003756	4334.994738	0.005817
-0.008771	14000.008474	0.000000	0.013302
-0.000000	0.000000	6999.994880	0.000000
-0.004726	6999.994880	14000.029383	0.005701
0.000000	14000.029383	0.000825	0.000000
4434.997976	0.000825	7000.005675	0.003197
4434.995540	7000.005675	7000.005675	0.010195
4434.994634	14000.009754	14000.009754	0.010404

6. 결론 및 추후 과제

본 연구는 선박 제작 과정 중의 자동화를 추진하기 위한 과정 중 선각 데이터 해석과 관련하여 다음의 성과를 얻었다.

- 3차원 측정 장치인 MONMOS의 측정 데이터를 PC로 전송하여 파일로 저장하는 프로그램을 개발하여 측정 데이터의 컴퓨터 관리가 가능하고, 설계 데이터와 함께 이용하여 해석 및 시뮬레이션을 가능케 하였다.
- 윈도우 그래픽 화면에서 계측치를 해석, 평가하는 프로그램을 개발하여 각 블록의 검사 데이터에서 가상 조립의 시뮬레이션 실험을 컴퓨터로 할 수 있게 함으로써 사전에 불량부분에 수정을 가해 차후의 조립 작업에 드는 시간을 단축할 수 있다.

선박의 건조 과정에서 실제로 사용하는 여러 블록의 설계치 및 측정치 파일의 부족으로 충분한 시뮬레이션을 해보지 못한 것이 아쉽다. 이상과 같이 시스템의 원형(prototype)을 개발하였으나 CAD 설계 데이터 관련 부분이 구현되지 않아 아직 현장 적용은 하지 못한 상태이다. 차후 현장 적용과 아울러 본 시스템을 개선해 사용함으로써 선박 제작 과정의 생산성 향상에 기여할 수 있으면 좋겠다.

참 고 문 헌

1. SDR4C Control Terminal Operation Manual, SOKKIA Korea
2. “데이터 통신”, Larry Hughes 원저, 조정완, 맹승렬, 윤현수 공역, 정익사, 1993
3. 이상규, 김지훈, “HHI 선각 CAD/CAM 일관 시스템 구축”, 기술현대 Vol 12, no 4, 1992
4. 한순홍, “STEP 표준의 근황과 선박 CAD 데이터 교환”, CAD & 그래픽스, 1995년 2월호, pp254--258
5. 요자끼 마사아끼 외 3인, “조선용 3차원 계측 해석 시스템의 개발과 그 적용”, 석천 도파마기보, 1991년 9월 제31권 제5호, pp365--370
6. 3D Analysis System User's Manual, Ishikawajima System Technology, 1993
7. “High-Performance C Graphics Programming for Windows”, Lee Adams, Windcrest/McGraw-Hill, 1992
8. “Windows Visualization Programming with C/C++ -- 3D Visualization, Simulation, and Virtual Reality”, Lee Adams, Windcrest/McGraw-Hill, 1994
9. “Computer Graphics”, Donald Hearn, M. P. Baker, Prentice-Hall, 1986
10. “Mathematical Elements for Computer Graphics”, David F. Rogers, J. A. Adams, 2nd edition, McGraw-Hill, 1990
11. B. Sabata and J. K. Aggarwal, “Estimation of motion from a pair of range images: A review”, CVGIP: Image Understanding, 54(3): 309--324, November 1991
12. R. Y. Tsai and T. S. Huang, “Uniqueness and estimation of three-dimensional motion parameters of rigid objects with curved surfaces”, IEEE Trans. on PAMI, PAMI-6(1): 13--27, Jan 1984
13. S. Umeyama, “Least-squares estimation of transformation parameters between two point patterns”, IEEE Trans. on PAMI, PAMI-13(4): 376--380, April 1991