

2차원 캐릭터의 입체화

소속

경상대학교 수학과

이름

신영민 정예진 김성윤 정재훈

목차

I. 수행 개요

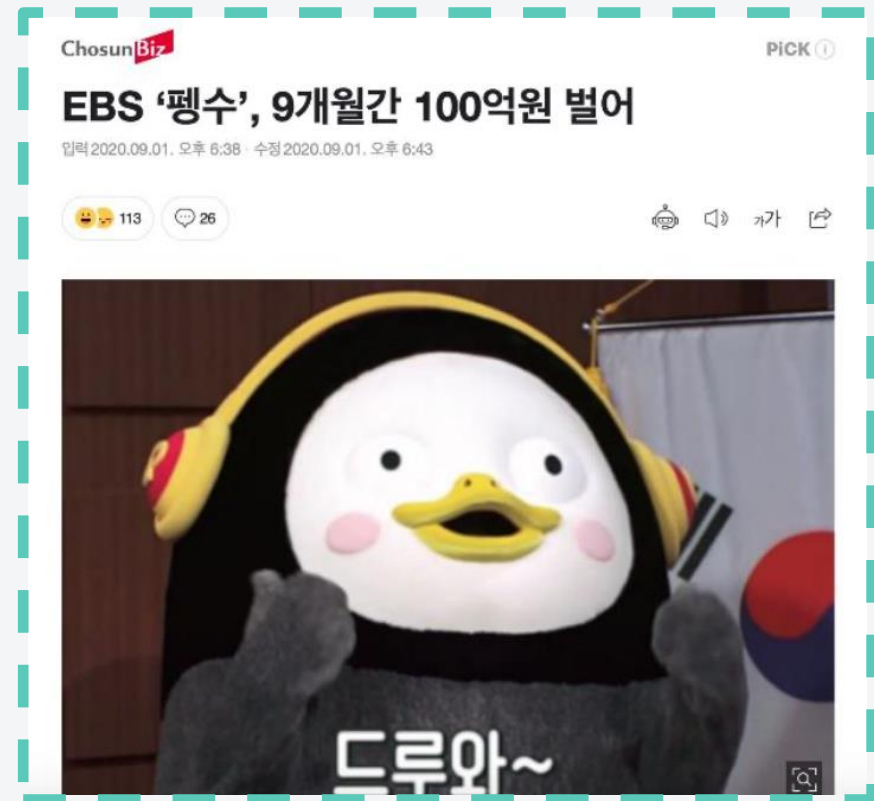
II. 수행 계획

III. 수행 과정

IV. 결론

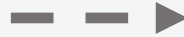
1. 수행 개요

- 캐릭터가 가진 산업적 가치

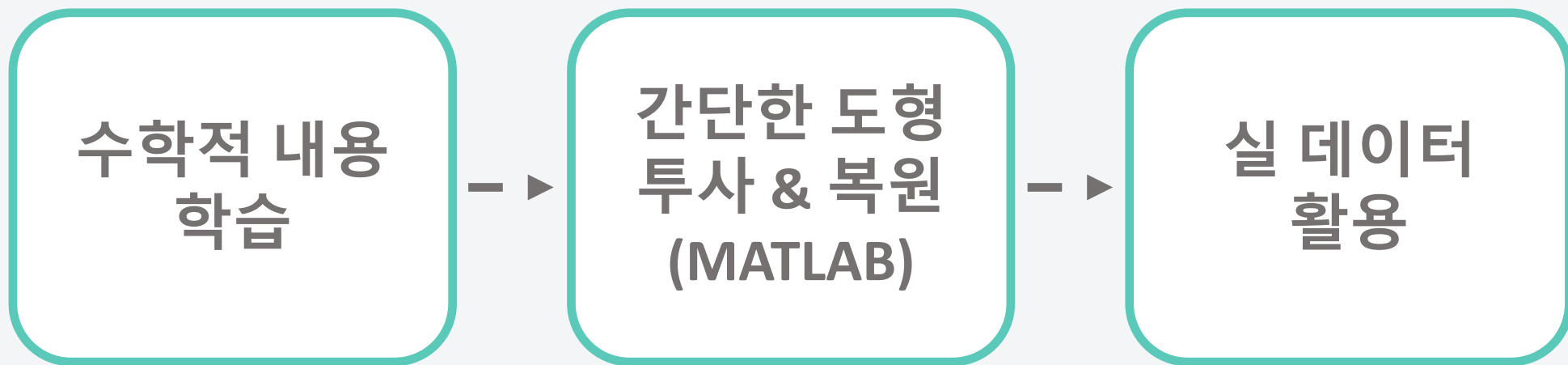


1. 수행 개요

- 캐릭터가 가진 산업적 가치



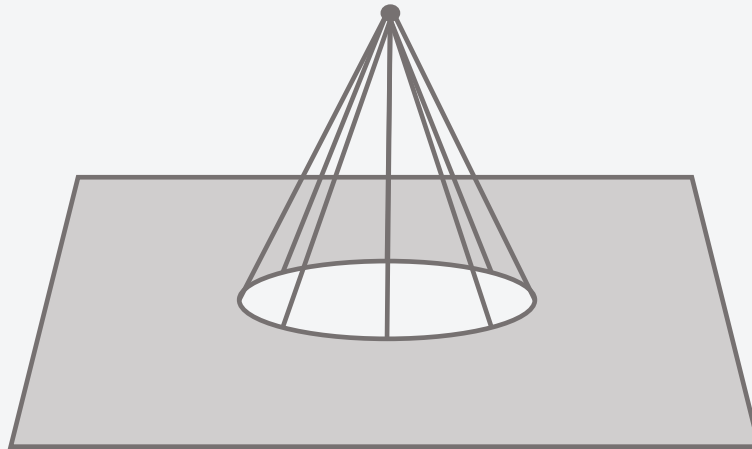
2. 수행 계획



3. 수행 과정_수학적 내용

- 사영

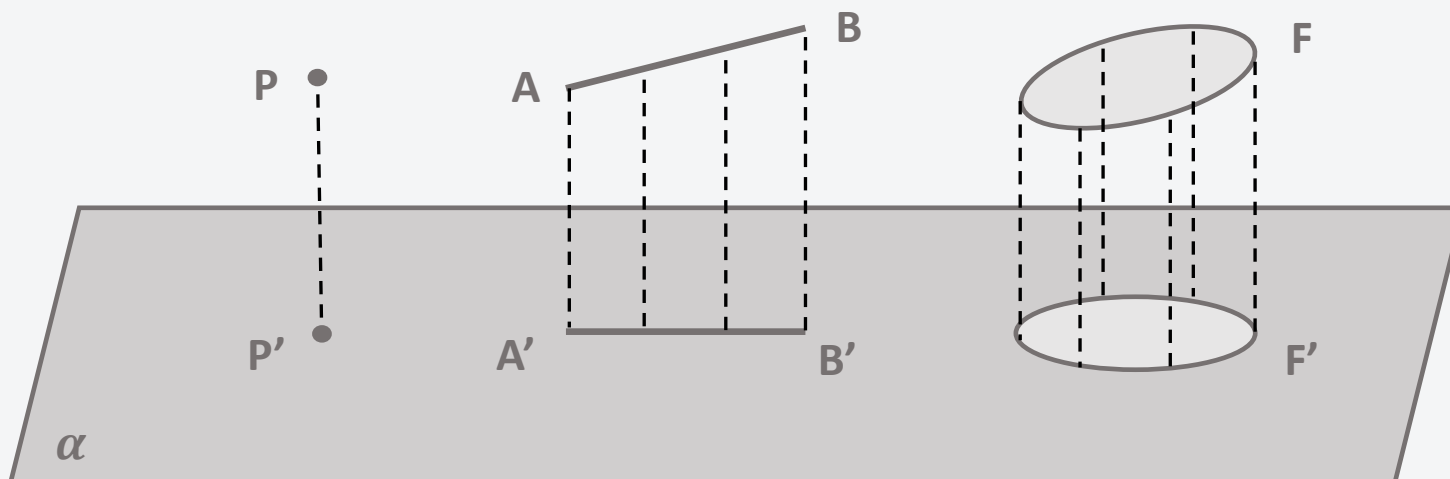
평면이나 공간 위에 있는 도형의 각 점을 평면이나 공간 위에 있지 않는 점을 잇는 직선을 긋는 것을 말합니다.



3. 수행 과정_수학적 내용

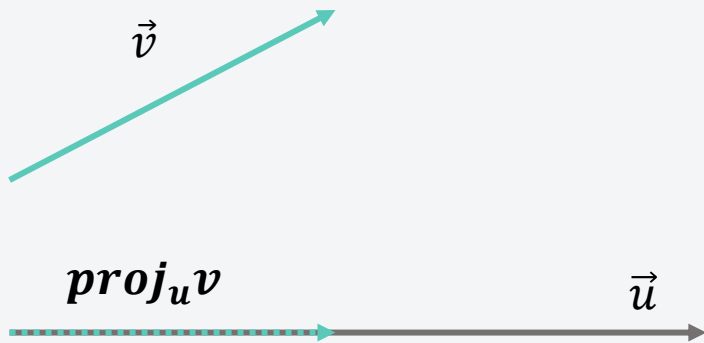
• 정사영

평면 α 밖의 점 P 에서 α 에 그은 수선의 발 P' 를 점 P 의 평면 α 위로의 정사영이라 한다. 또, 도형 F 에 속하는 모든 점의 평면 α 위로의 정사영으로 이루어지는 도형 F' 을 F 의 α 위로의 정사영이라고 한다.



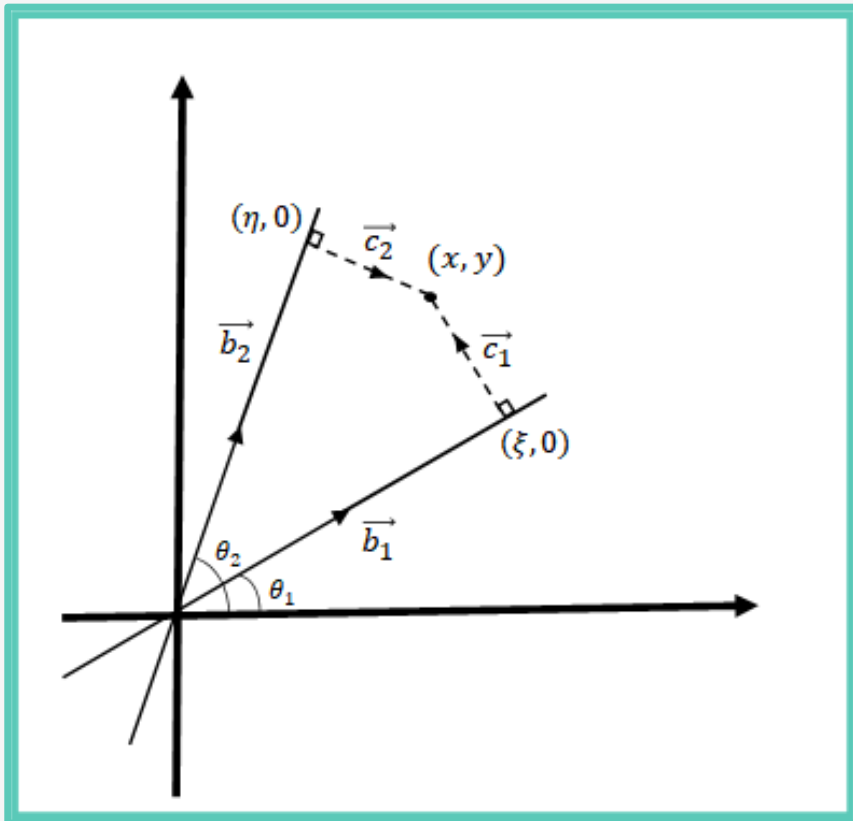
3. 수행 과정_수학적 내용

- $proj_u(v) : \vec{u}$ 위로의 \vec{v} 의 정사영
$$proj_u(v) = \frac{\vec{u} \cdot \vec{v}}{\vec{u} \cdot \vec{u}} \vec{u}$$



3. 수행 과정_점과 직선

- 점 $\mathbb{X} = (x, y)$ 를 \vec{b}_1 와 \vec{b}_2 로의 정사영

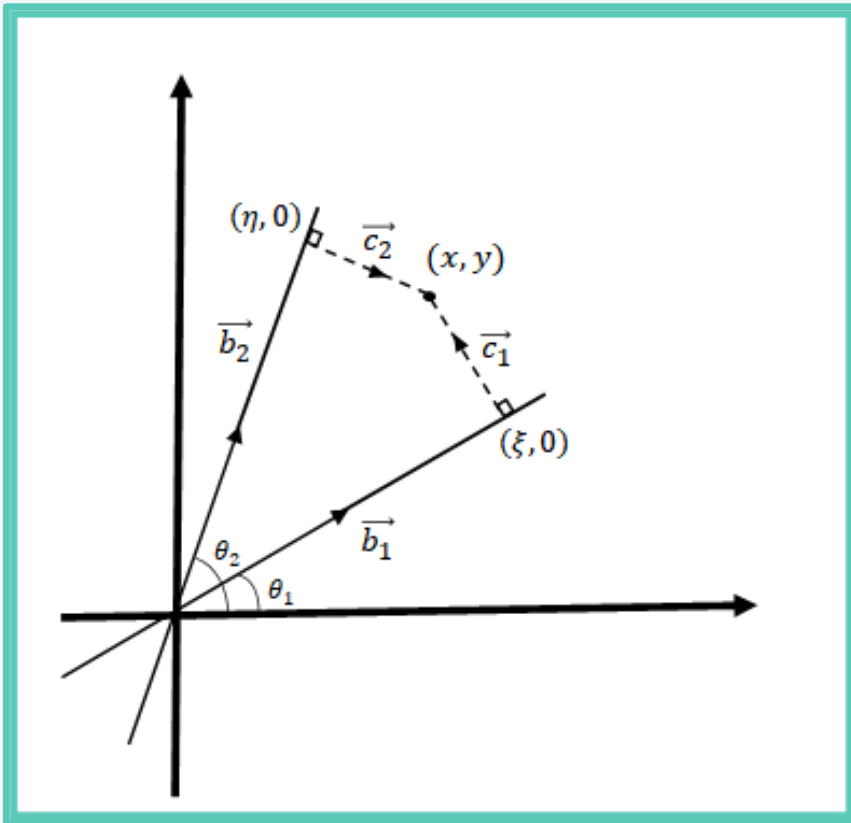


$$(1) \quad \left\| \text{proj}_{\vec{b}_1}(\mathbb{X}) \right\| = \xi$$

$$(2) \quad \left\| \text{proj}_{\vec{b}_2}(\mathbb{X}) \right\| = \eta$$

3. 수행 과정_점과 직선

- 정사영한 점의 복원



$$\vec{b_1} \perp \vec{c_1}, \vec{b_2} \perp \vec{c_2}$$

$$\mathbb{X} - \xi \vec{b_1} = \vec{c_1}$$

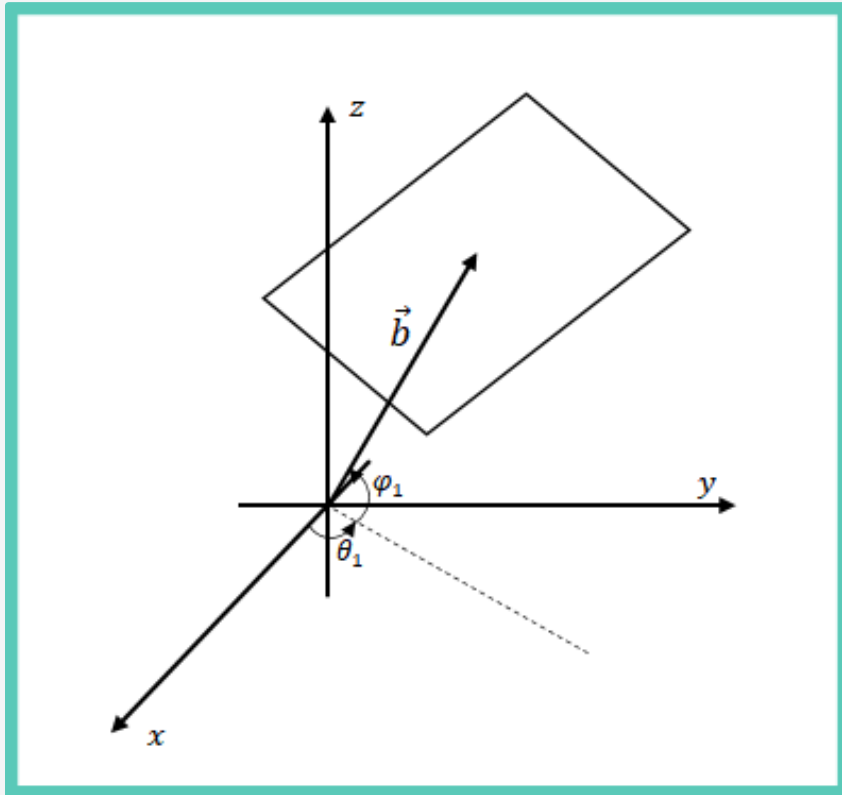
$$\mathbb{X} - \eta \vec{b_2} = \vec{c_2}$$

$$\Rightarrow \vec{b_1} \cdot \mathbb{X} = \xi \|\vec{b_1}\|, \quad \vec{b_2} \cdot \mathbb{X} = \eta \|\vec{b_2}\|$$

$$\dashrightarrow \begin{bmatrix} \cos(\theta_1) & \sin(\theta_1) \\ \cos(\theta_2) & \sin(\theta_2) \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \xi \|\vec{b_1}\| \\ \eta \|\vec{b_2}\| \end{bmatrix}$$

3. 수행 과정_점과 평면

- 점 $\mathbb{X} = (x, y)$ 를 평면으로의 정사영

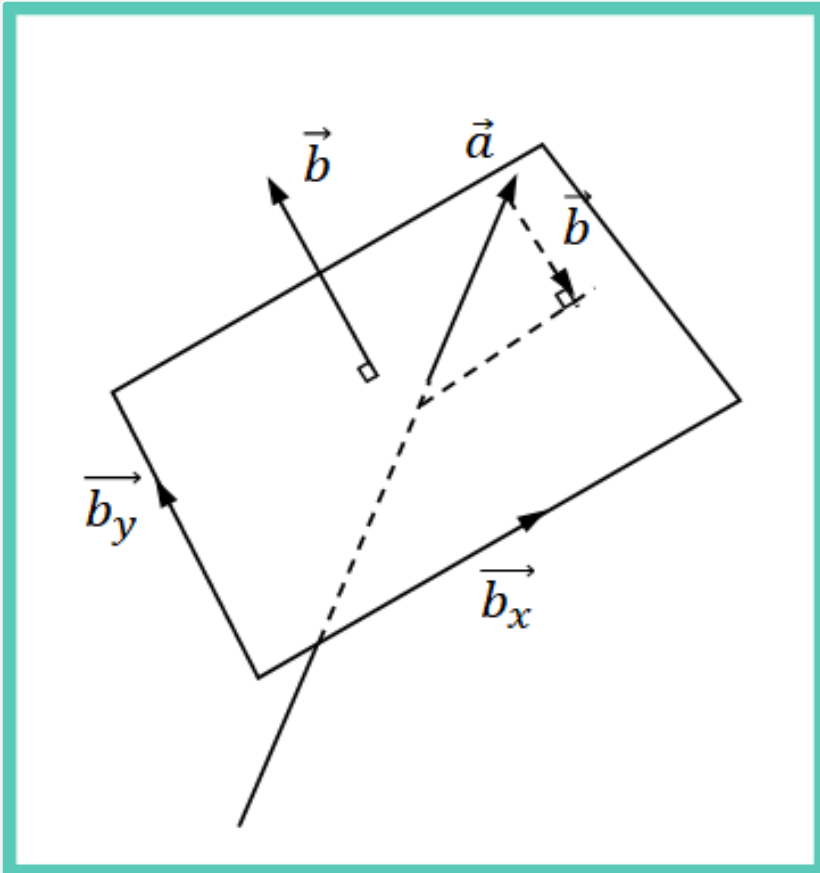


$\vec{b} \cdot \mathbb{E} = 0$: \mathbb{E} 는 \vec{b} 에 수직인 평면

$$\vec{b} = (\cos(\theta_1) \cos(\varphi_1), \sin(\theta_1) \cos(\varphi_1), \sin(\varphi_1))$$

3. 수행 과정_점과 평면

- 점 $\mathbb{X} = (x, y)$ 를 평면으로의 정사영



$\vec{b} \cdot \mathbb{E} = 0$: \mathbb{E} 는 \vec{b} 에 수직인 평면

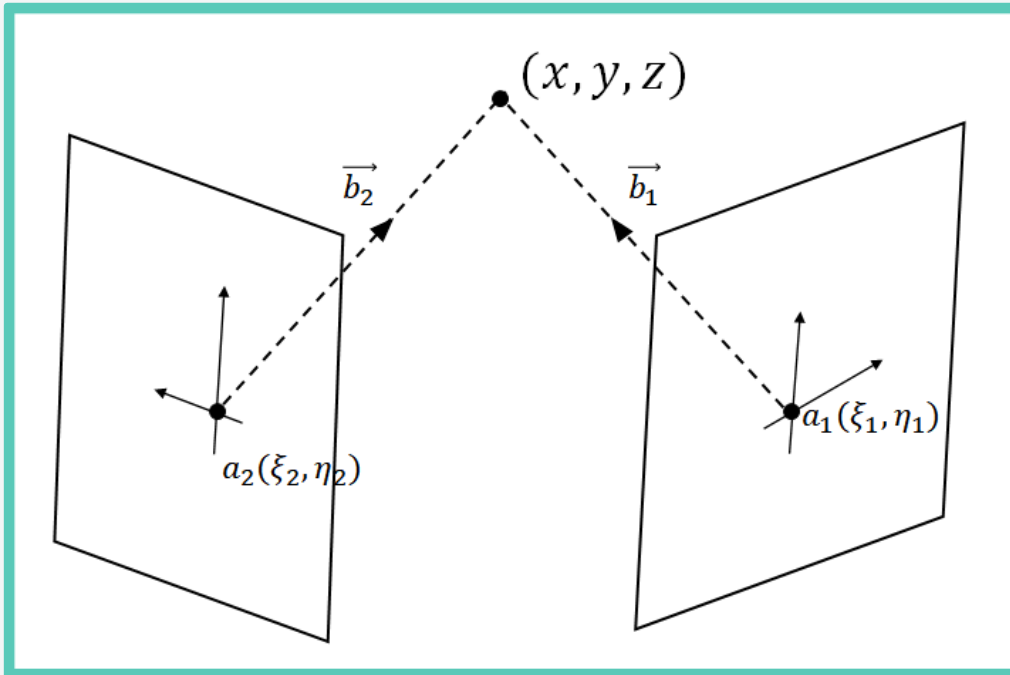
$$\vec{b} = (\cos(\theta_1) \cos(\varphi_1), \sin(\theta_1) \cos(\varphi_1), \sin(\varphi_1))$$

$$\vec{b}_x = (-\sin(\theta_1 + \varphi_1), \cos(\theta_1 + \varphi_1), \cos(\varphi_1))$$

$$\vec{b}_y = \vec{b} \times \vec{b}_x$$

3. 수행 과정_점과 평면

- 정사영한 점의 복원



(1-x)

$$\|proj_{\vec{b}_1^x}(\mathbb{X})\| = \xi_1$$

(1-y)

$$\|proj_{\vec{b}_2^y}(\mathbb{X})\| = \eta_1$$

(2-x)

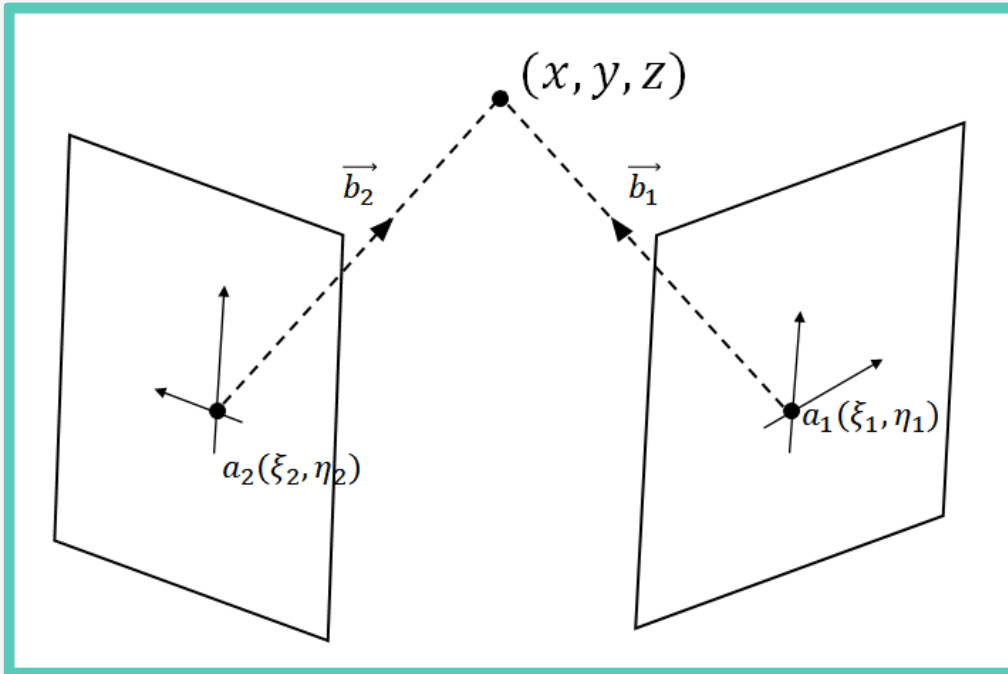
$$\|proj_{\vec{b}_2^x}(\mathbb{X})\| = \xi_2$$

(2-y)

$$\|proj_{\vec{b}_2^y}(\mathbb{X})\| = \eta_2$$

3. 수행 과정_점과 평면

- 정사영한 점의 복원



$$\vec{b}_1^x (x - \xi_1 \vec{b}_1^x) = 0 \quad \vec{b}_1^y (y - \eta_1 \vec{b}_1^y) = 0$$

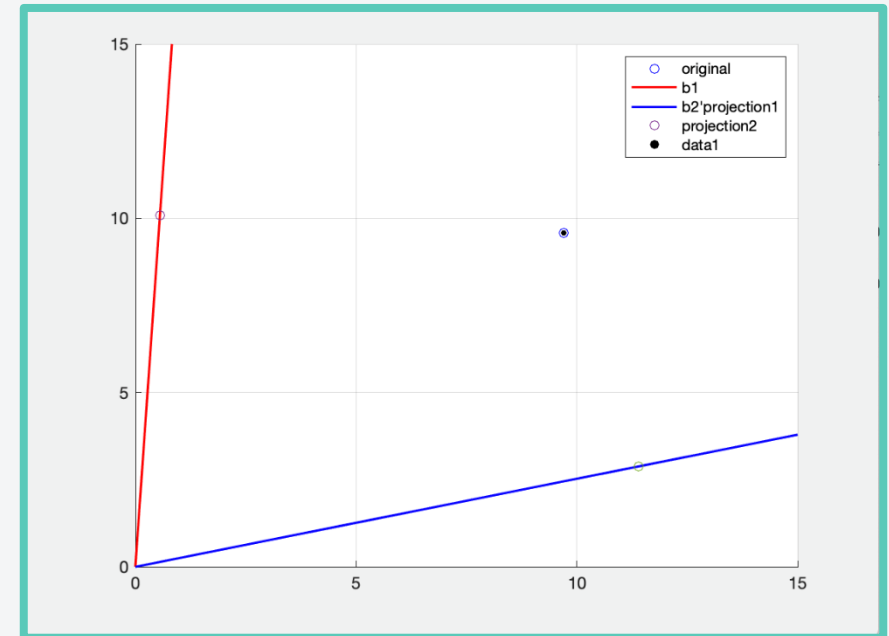
$$\vec{b}_2^x (x - \xi_2 \vec{b}_2^x) = 0 \quad \vec{b}_2^y (y - \eta_2 \vec{b}_2^y) = 0$$

$$\begin{bmatrix} \vec{b}_1^x \\ \vec{b}_1^y \\ \vec{b}_2^x \\ \vec{b}_2^y \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} \xi_1 \|\vec{b}_1^x\|^2 \\ \eta_1 \|\vec{b}_1^y\|^2 \\ \xi_2 \|\vec{b}_2^x\|^2 \\ \eta_2 \|\vec{b}_2^y\|^2 \end{bmatrix}$$

3. 수행 과정_MATLAB

• 점과 벡터

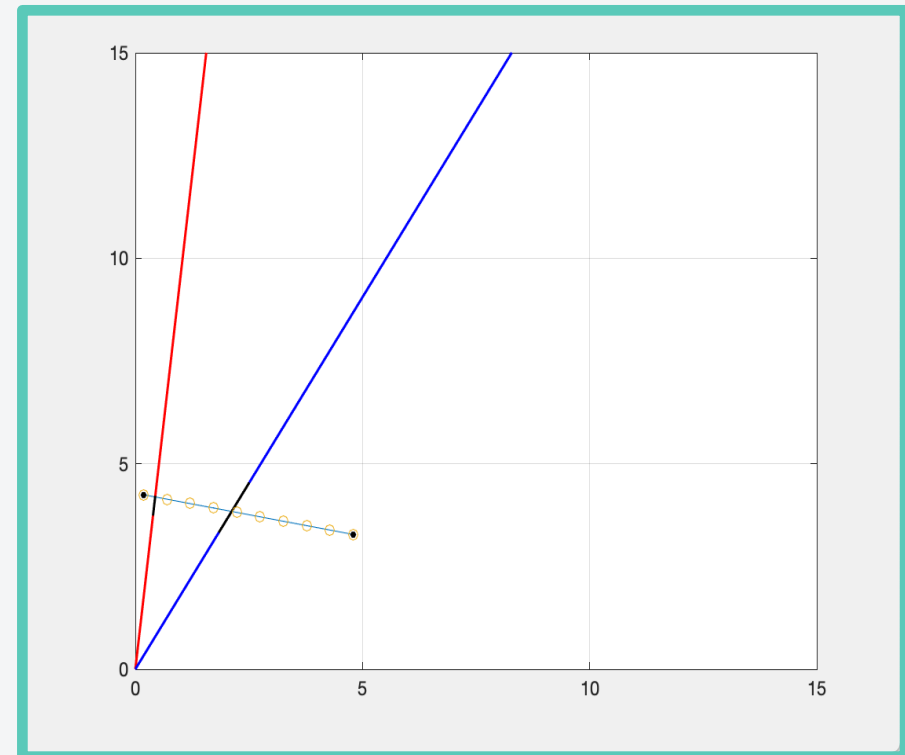
```
1 - clear; clc; close all;
2
3 - p=0:0.1:15; %2차원 좌표 만들기
4 - the1=rand*pi/2;
5 - the2=rand*pi/2;
6
7 - b1=sin(the1)/cos(the1);
8 - b2=sin(the2)/cos(the2);
9
10 - x=10*rand; %random x 설정
11 - y=10*rand; %random y 설정 (0에서 10사이의 값)
12 - scatter(x,y,'b') %original (x,y)좌표 찍어보기
13
14 - grid on
15 - axis([0 15 0 15]);
16 - hold on
17 - plot(p,p*b1,'r','Linewidth',1.5); %b1벡터의 직선 그리기
18 - hold on
19 - plot(p,p*b2,'b','Linewidth',1.5);
20
21 - c1=(x*sin(the1)-y*cos(the1));
```



3. 수행 과정_MATLAB

• 직선과 벡터

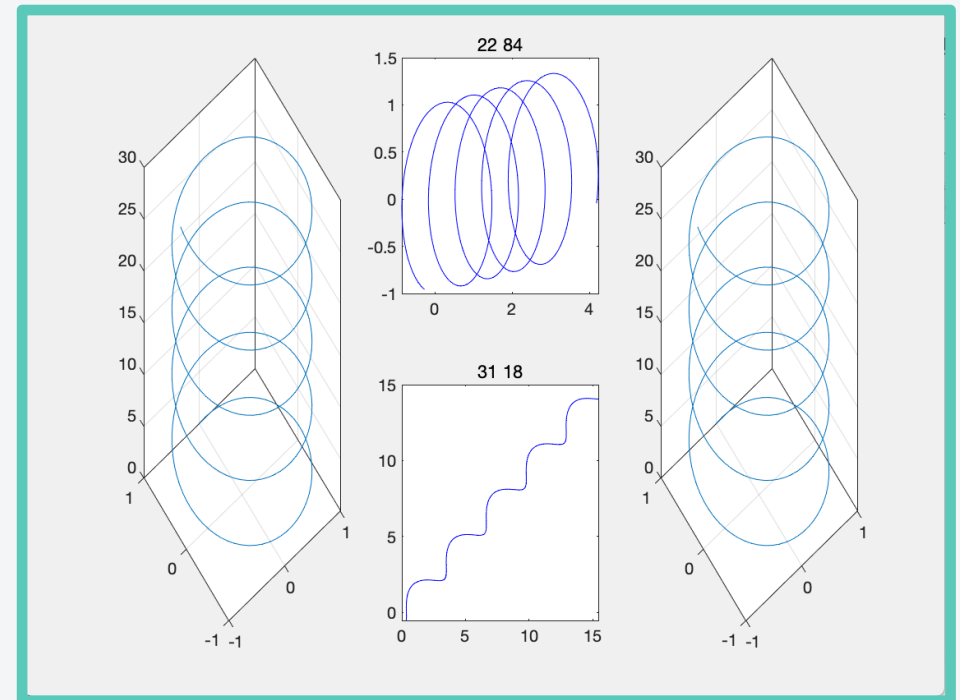
```
31  
32 — grid on  
33 — hold on  
34 — plot(p,p*b1,'r','Linewidth',1.5)  
35 — hold on  
36 — plot(p,p*b2,'b','Linewidth',1.5)  
37 — % return  
38 — |  
39 — % Projection %  
40 —  
41 — c1_1=(x1*sin(the1)-y1*cos(the1)); % Generate vector c1_1 for x1,y1  
42 — c1_2=(x2*sin(the1)-y2*cos(the1)); % Generate vector c1_2 for x2,y2
```



3. 수행 과정_MATLAB

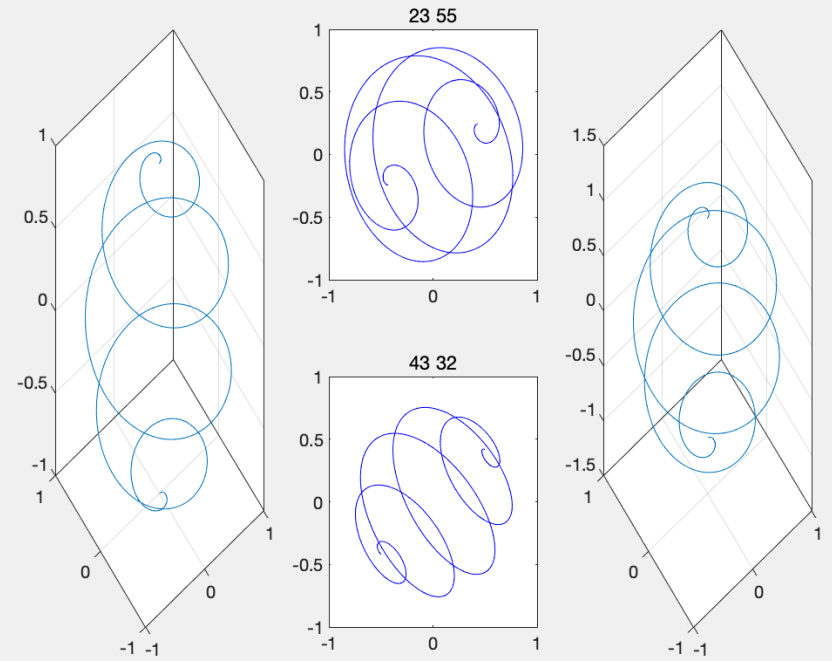
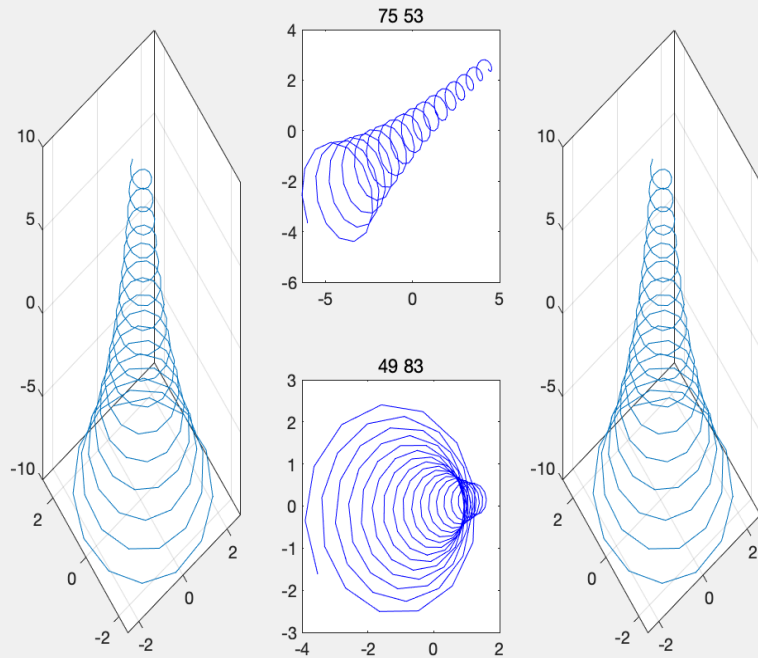
• 곡선과 평면

```
68  
69 — mDataX2=real*bx2/sum(bx2.^2);      % bx2 벡터에 투사한 좌표  
70 — mDataY2=real*by2/sum(by2.^2);      % by2 벡터에 투사한 좌표  
71  
72 — plot(mDataX2,mDataY2,'b');          % bx2 벡터와 by2 벡터의 좌표로 그리기  
73 — title(sprintf('%1.0f %1.0f',the2*180/pi,phi2*180/pi));  
74  
75 — subplot(2,3,[3,6])  
76  
77 — B=[bx1 by1 bx2 by2];  
78 — b=[sum(bx1.^2)*mDataX1 sum(by1.^2)*mDataY1 ...  
79 —     sum(bx2.^2)*mDataX2 sum(by2.^2)*mDataY2];  
80  
81 — make=B'#b';
```



3. 수행 과정_MATLAB

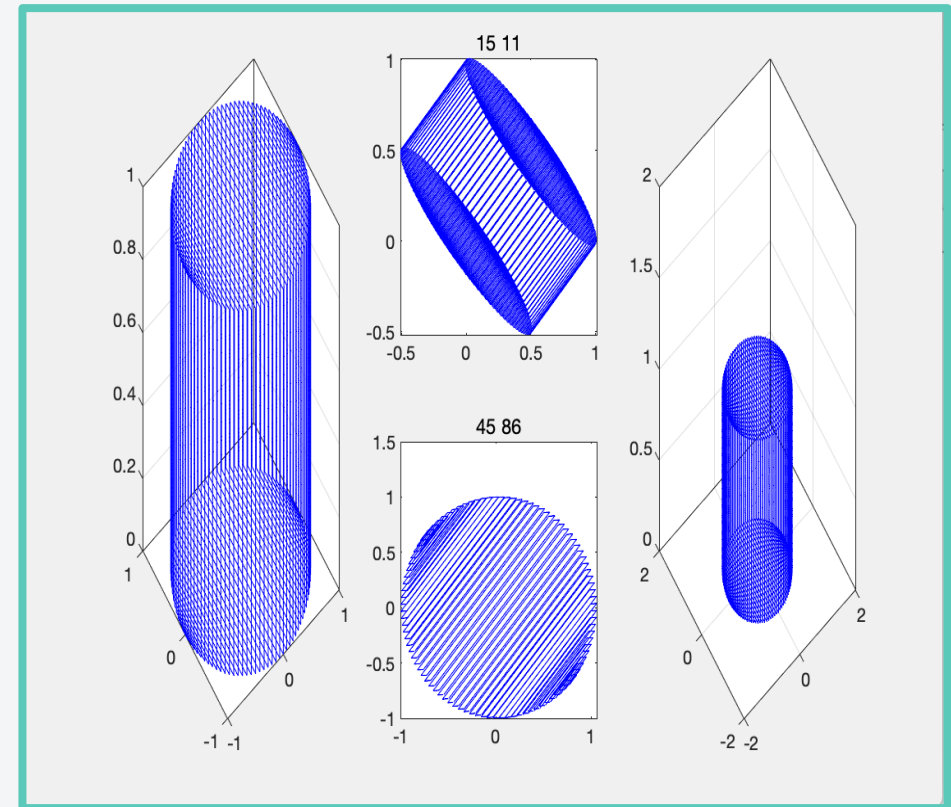
- 곡선과 평면



3. 수행 과정_MATLAB

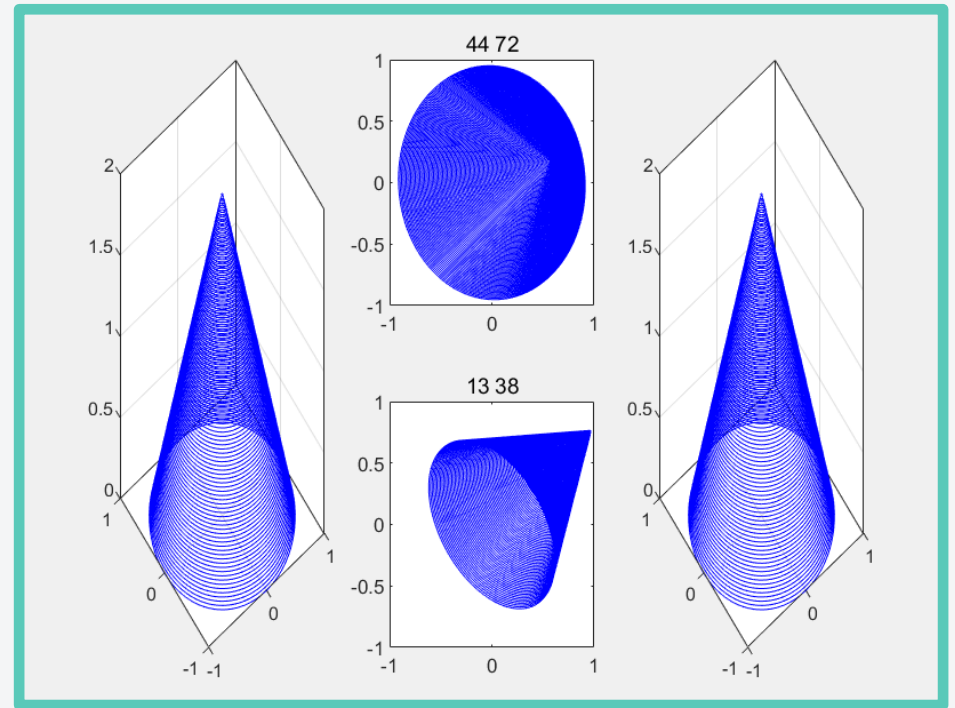
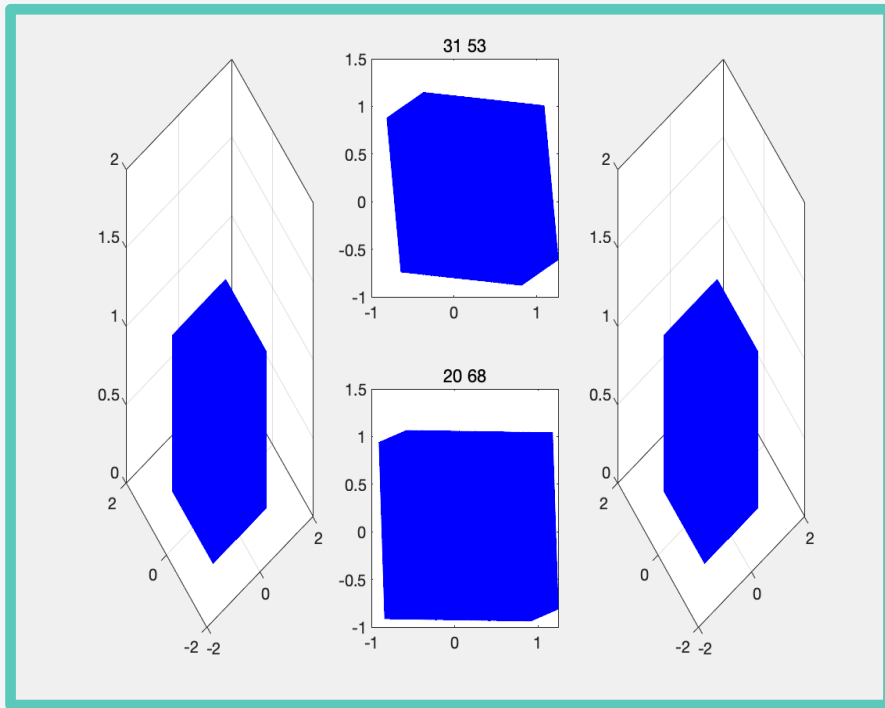
• 도형과 평면

```
1 clear; clc; close all;
2
3 [x,y,z]=cylinder(1,100);           % 반지름이 1인 원기둥 그리기
4
5 xx=x(1,:)' ;                       % 원기둥 x 데이터 받기
6 yy=y(1,:)' ;                       % 원기둥 y 데이터 받기
7
8 t=0:0.01:1;                         % 원기둥의 z 데이터 설정
9
10 the1=rand*pi/2;                     % the1 각
11 phi1=rand*pi/2;                     % phi1 각
12
13 b1=[cos(the1)*cos(phi1)
14     sin(the1)*cos(phi1)
```



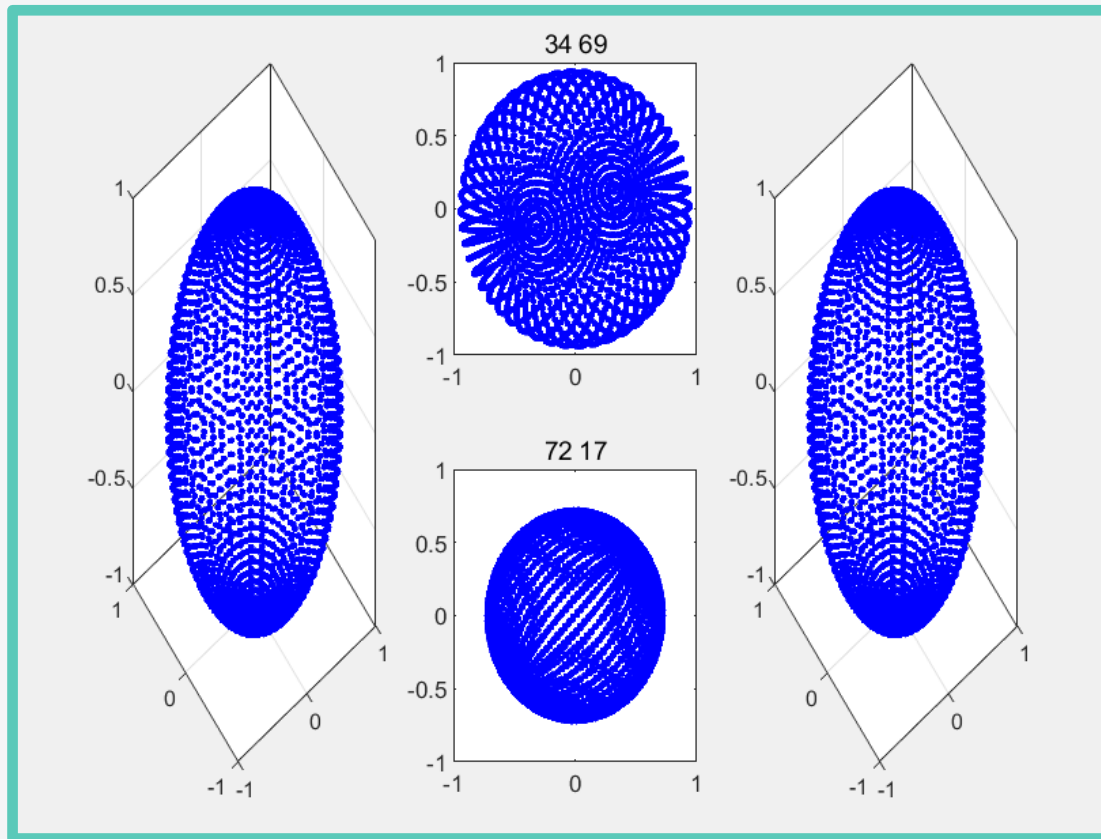
3. 수행 과정_MATLAB

- 도형과 평면



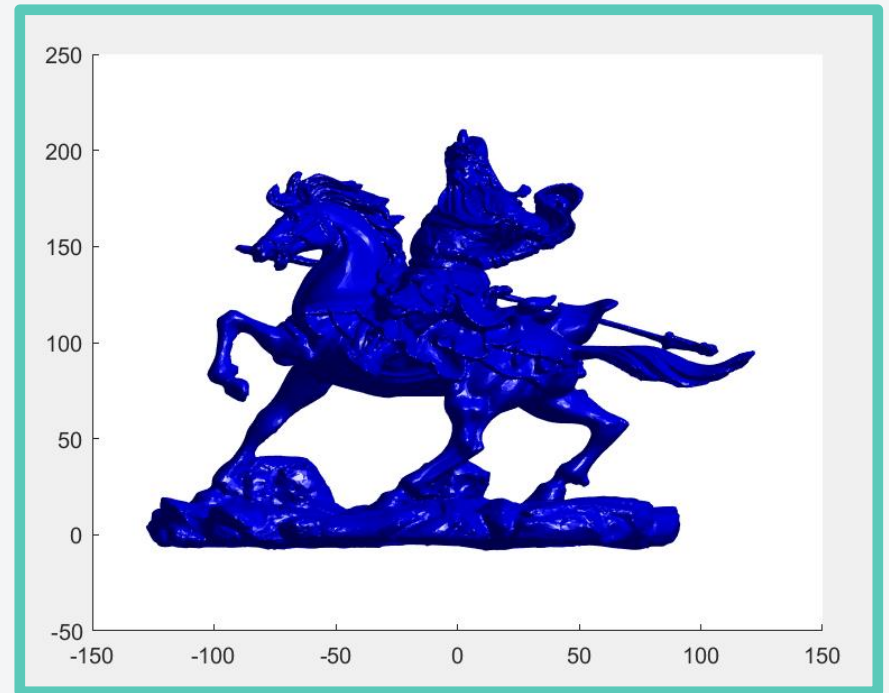
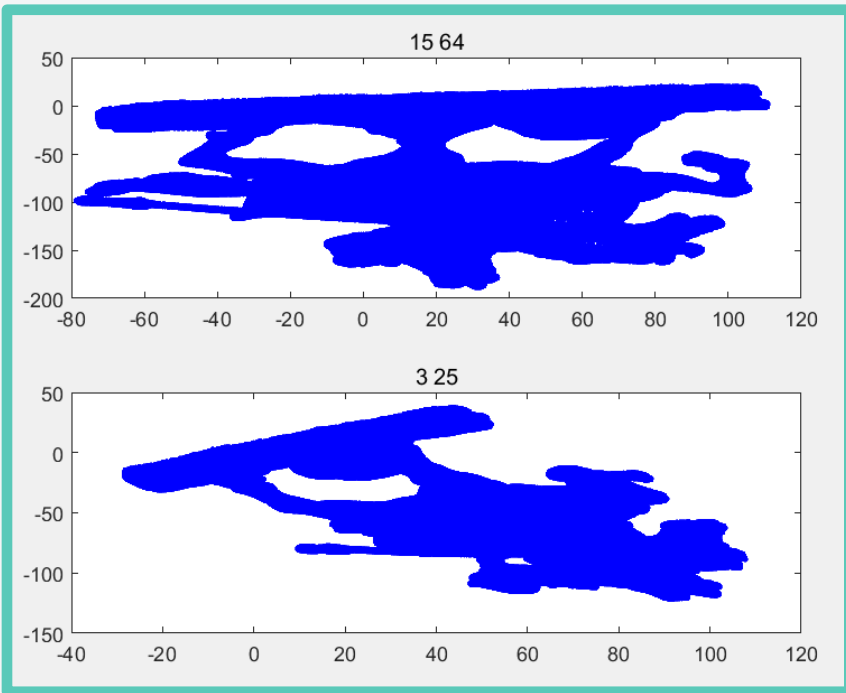
3. 수행 과정_MATLAB

- 도형과 평면(구)



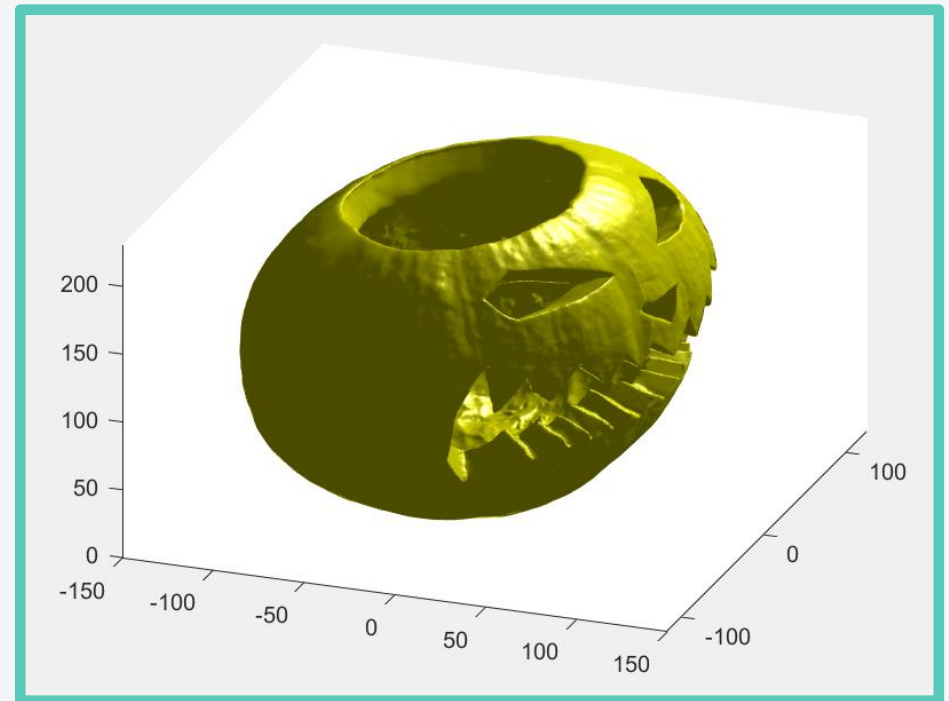
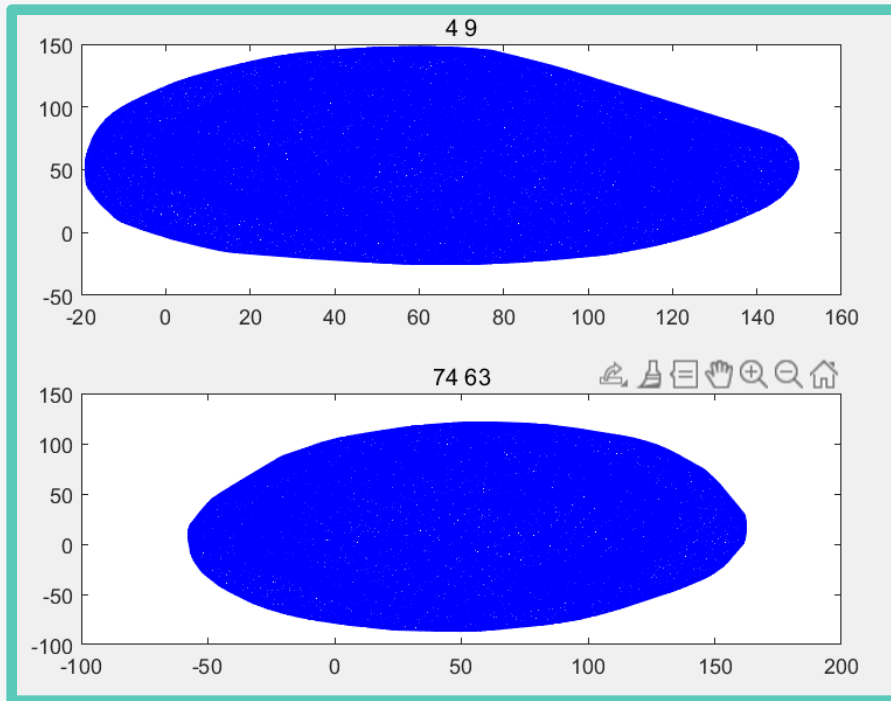
3. 수행 과정_MATLAB

- stl (청동 조각상)



3. 수행 과정_MATLAB

- stl (Pumkin)



4. 결론

2차원 도형
→ 3차원 도형

2차원 캐릭터
→ 3차원 캐릭터

산업적 이용
가능성

Q&A
