Berlin, Germany

Three dimensional coordinates into two dimensional coordinates transformation

Edward Gerhold

July 9, 2015

Version 0.2.6 (the very much a draft paper)
Has to be ordered, corrected, mathematically upgraded and to get more pictures.
This is an active working draft. Which is now totally out of its shape this version.

Contents

1 Introduction

On a piece of paper you see three coordinate axes pointing into three directions in space. In reality these vectors are two dimensional. Because they point into three directions on the paper, and not into the real space.

In this document we will design a basis for the coordinate transformation. A basis is multiplied with the value of the coordinate to move to the correct new point.

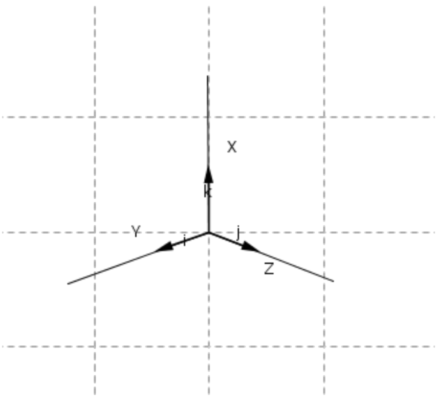

Figure 1: Picture of a 3-D coordinate system with ijk-vectors on the axes pointing into three directions. See [1] for introduction.

In the case of cosines and sines, we move left and right and up and down, to tell you directly, what happens, when we multiply the coordinates with the matrix.

What we will do in the document

1. Choose angles for our coordinate axes around the unit circle to lay out three axes.
2. Write down the basis vectors for each coordinate axis
3. Assemble a matrix with the vector basis for a point by point transformation.
4. Read the example source code for a computer function, which is exactly two lines long. One for the new x and one for the new y.
5. Derive the generic case of transforming coordinate systems down to the plane.

2 Definitions

2.1 Definition of the angles for the coordinate axes

Let φ_n be the set of axis angles, one for each axis. The angles start at the same place, at the number zero. You have to arrange the x, y, and z axes like on a piece of paper around the unit circle by giving them the appropriate angles. All three angles start at the default at zero at the horizontal axis of the plane.

$$\varphi_n := \{\varphi_x, \varphi_y, \varphi_z\}$$

Example The function rad converts degrees to radians, its useful for computer functions taking radians.

$$\text{rad}(\phi) := \frac{\pi}{180} \times \phi, \phi \in R$$

Here is an example of three angles. The three axes have an angle of 120 degrees between each.

$$\varphi_x = \text{rad}(210), \varphi_y = \text{rad}(330), \varphi_z = \text{rad}(90)$$

$$\varphi_x = \frac{\pi}{180} \times 210 = \frac{7\pi}{6}, \varphi_y = \frac{\pi}{180} \times 330 = \frac{11\pi}{6}, \varphi_z = \frac{\pi}{180} \times 90 = \frac{\pi}{2}$$

Calculating the three values is fun. More fun with angles could be had with [2]. What the values are good for? The angles have to be passed to the cosine and sine functions, when we setup the basis vectors around a circle.

2.2 Definition of the three 2-D basis vectors

Let e_n be the set of three two dimensional basis vectors, namely $\vec{e}_x$, $\vec{e}_y$ and $\vec{e}_z$. Other names are $\vec{i}$, $\vec{j}$ or $\vec{k}$ for example, like on the picture of the coordinate system mentioned. The three vectors point into the three directions of the three axes. On a piece of paper they are two dimensional, becausee they point into three directions on the paper. The space being shown is what our brain completes, seeing three correct axes. The three basis vectors represent exactly one unit into the direction of each axis. This unit can be manipulated by multiplying the vector components inside e_n with a factor r_n.

$$e_n := \{\vec{e}_x, \vec{e}_y, \vec{e}_z\}$$

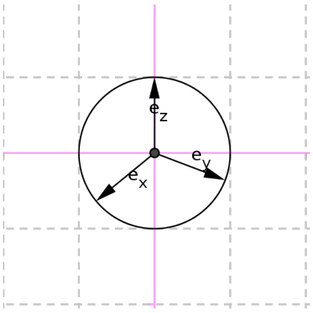

Figure 2: The three basis vectors point into the positive directions of the desired coordinate axis. They are arranged around a circle with the trigonometric functions of cosine and sine.

This is the set of three basis vectors in set notation. To guess no numbers, its easier for us, for each vector, to go around the unit circle by the angles we already defined and to use cosine and sine for the correct x-distance and y-distance. For help, you should remember this parametrization of x and y from the unit circle. [1]

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} r\cos\varphi \\ r\sin\varphi \end{pmatrix}$$

Alternativly it can be written like

$$(x, y) = (r\cos\varphi, r\sin\varphi)$$

Increasing r increases the radius of the circle and the distance of the point and the length of the vector r. Its called the unit circle when $r = 1$.

Modeling the three two dimensional basis vectors with this information, we get the following three two dimensional basis vectors. They point along the coordinate axes and are the ruler for our transformation.

$$\vec{e}_x := (r_x\cos(\varphi_x), r_x\sin(\varphi_x))^T = \begin{pmatrix} r_x\cos(\varphi_x) \\ r_x\sin(\varphi_x) \end{pmatrix}$$
$$\vec{e}_y := (r_y\cos(\varphi_y), r_y\sin(\varphi_y))^T = \begin{pmatrix} r_y\cos(\varphi_y) \\ r_y\sin(\varphi_y) \end{pmatrix}$$
$$\vec{e}_z := (r_z\cos(\varphi_z), r_z\sin(\varphi_z))^T = \begin{pmatrix} r_z\cos(\varphi_z) \\ r_z\sin(\varphi_z) \end{pmatrix}$$

[1] Interested people can read about parametrization of x and y, the unit circle, polar coordinates and cosine and sine for example in the books [1], [2], [4].

One for each component of (x, y, z) By multiplying with, we move by the points into the right directions. On the plane we use to point into three directions. This means that each component contributes a move. And the final position is the right position of the new coordinate.

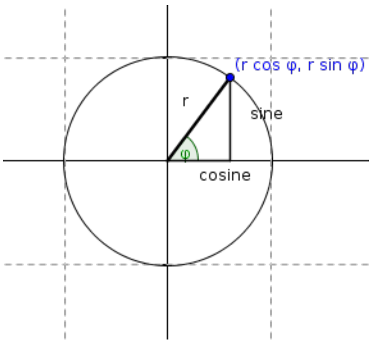

Figure 3: A picture of the unit circle, the hypotenuse r, the adjacent cosine, the opposite sine and the angle φ. It is a circle of radius r, and no longer the unit circle, if $r \neq 1$.

Remark.The values of r_x, r_y and r_z decide, how long one unit into each direction is. To preserve affine graphical transformations all three axes should have the same length, to represent same distance for each coordinate unit. The units can generally be enlarged or made smaller than *unit length*, which is a length of 1. By default the resulting vectors with the cosine for x and sine for y components have *unit length*, if you dont multiply the cosine and the sine in each basis vector with r_x, r_y and r_z. The r_n represent the lengths of the hypotenuses or the new radius of the circle around the origin described by the vectors.

Remark On the other side, the length of r can be determined from existing basis vectors. By pulling the square root out of the sum of the squares of the vector components. This is also known as euclidean norm, or the root of the inner product of the vector with itself. Like real fans of sines and cosines, we know that $\sin^2 \varphi + \cos^2 \varphi = 1$ and what the root of 1 is. If we pull the root out of the products of cosine and sine multiplied with $r \neq 1$, we get the length of r again. $r = \sqrt{\vec{x} \cdot \vec{x}} = \sqrt{(\vec{x}, \vec{x})} = \left(\Sigma_{i=1}^{n} \vec{x}_i^2\right)^{\frac{1}{2}} = \|\vec{x}\|$

2.3 Lemma of the orthogonal bases

Remark. Not corrected in this version, i can use the word *basis* together with the formula without the adjective. New lecture scripts proved me. The orthogonal case is then taught a paragraph later.

The one lemma we need is the generic theorem for multiplying a vector with some basis for some coordinate system.

In our case the orthogonality by 90 degrees rules do not count. The basis vectors are no longer perpendicular. Or in other words $\neg (\vec{e}_i \perp \vec{e}_j)$ with $i, j \in \{x, y, z\}$.

The plane gives us two degrees of freedom[2], to go horizontal or vertical. And in a cartesian coordinate system with infinite points, we can choose any direction around a point (x,y). Any not straight move will go horizontally or vertically by componentwise amounts. Any straight move will go by one of the components only. We arrange for example with about 120 degrees[3] between each axis around the origin on a plane.

The point is, the generic formula still holds.
The formula for multiplying a vector with a base to get a new vector is this.[4]

[2] A version later i am asking myself, if i may use degree of freedom? According to https://de.wikipedia.org/wiki/Freiheitsgrade. Maybe i have explained it understandable, but wrong now.

[3] The angle between e_x and e_y can be a little bigger, if e_z points up. I have noticed this on pictures. And before you get crazy, you are allowed to choose any angles. Let x,y be perpendicular or smaller. It is your choice.

[4] The formula can be found in many mathematics, chemistry and physics lecture scripts, and a good introduction is [3].

$$\vec{v} = \sum_{i=1}^{n} \vec{x}_i \vec{e}_i$$

It is done componentwise for each row of the vector. n is the number of the source dimensions. In our case it is $n = 3$. We are summing three products for each component of the new vector. Our old $\vec{x}$ is a $\vec{x} \in R^3$.
With $\vec{x}_i$ as the coordinate component and $\vec{e}_i$ as the corresponding basis vector and the current component. $\vec{v}$ is the resulting new vector. The new vector $\vec{v}$ is a $\vec{v} \in R^2$.

This is also equal to

$$\vec{v} = x\vec{i} + y\vec{j} + z\vec{k}$$

what also explains, what the ijk-Notation means. If you dont use it already for determining determinants for calculating cross products. It is for describing a vector. Dont forget, our i, j, k basis is two dimensional, because we draw on a 2-D plane like the computer screen or a piece of paper.

With a 3x3 basis the vector $x\vec{i} + y\vec{j} + z\vec{k}$ is equal to $\begin{pmatrix} x' \\ y' \\ z' \end{pmatrix}$. But with a 2x3 basis the vector $x\vec{i} + y\vec{j} + z\vec{k}$ is becoming $\begin{pmatrix} x' \\ y' \end{pmatrix}$

2.4 Finishing the matrix

Each (x, y, z) coordinate has to be multiplied for the new (x', y') with its corresponding term of the basis vectors in the matrix. That means, to sum the products with (x, y, z) and the cos terms up for x' and to sum the products of (x, y, z) and the sin terms up for y'. This is the same as imagining walking left and right with $\cos\varphi$ and up and down with $\sin\varphi$. Or mathematically adding positive or negative values.

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} xr_x \cos(\varphi_x) + yr_y \cos(\varphi_y) + zr_z \cos(\varphi_z) \\ xr_x \sin(\varphi_x) + yr_y \sin(\varphi_y) + zr_z \sin(\varphi_z) \end{pmatrix}$$

Remark The sum of the basis multiplied with the coordinates is nothing new. But literature and lecture scripts just tell how to multiply same dimensions, giving no clue about the easy 3-D to 2-D conversions. I can not speak for all, and do not believe, that no one told no one about this, but i have not met any one in the last twenty years telling me about this easy transformation and computer graphics went another, more complicated way, over homogeneous coordinates, 4 by 4 matrices and a final viewport division, to get the points back to two dimensions.

3 Theorem

3.1 The transformation matrix

Definition 1 *Let $\boldsymbol{A}$ be the matrix containing the three, two dimensional and trigonometric, basis vectors in order, one each column. You get a rectangular 2x3 matrix $\boldsymbol{A} \in R^{2x3} : R^3 \to R^2$. With the basis vectors $\begin{pmatrix} r_n \cos\varphi_n \\ r_n \sin\varphi_n \end{pmatrix}$ in the three columns.*

$$\boldsymbol{A} := \begin{pmatrix} \vec{e}_x & \vec{e}_y & \vec{e}_z \end{pmatrix} = \begin{pmatrix} r_x \cos(\varphi_x) & r_y \cos(\varphi_y) & r_z \cos(\varphi_z) \\ r_x \sin(\varphi_x) & r_y \sin(\varphi_y) & r_z \sin(\varphi_z) \end{pmatrix}$$

$\boldsymbol{A}$ should be treated as a function $\boldsymbol{A} \in R^{2x3} : R^3 \to R^2$. $(\vec{x}) \mapsto \boldsymbol{A}\vec{x}$.

3.2 The transformation

Theorem (The Fundamental Theorem of transforming 3-D Points into 2-D Points) 1 *If you multiply $\boldsymbol{A}$, the matrix of the three two-dimensional basis vectors, with the three-coordinate point (x, y, z), the result is a two coordinate point, (x', y'). This point (x', y') is the correct point on the two dimensional plane, representing the point (x, y, z) from the three dimensional coordinate system, you are transforming.*

$$\boldsymbol{A}\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} x' \\ y' \end{pmatrix}$$

Applying the operator performs the following operation

$$\boldsymbol{A}\begin{pmatrix} x \\ y \\ z \end{pmatrix} = x\vec{e}_x + y\vec{e}_y + z\vec{e}_z = \begin{pmatrix} xr_x\cos(\varphi_x) + yr_y\cos(\varphi_y) + zr_z\cos(\varphi_z) \\ xr_x\sin(\varphi_x) + yr_y\sin(\varphi_y) + zr_z\sin(\varphi_z) \end{pmatrix} = \begin{pmatrix} x' \\ y' \end{pmatrix}$$

Proof

$$\boldsymbol{A}\begin{pmatrix} x \\ y \\ z \end{pmatrix} = x\vec{e}_x + y\vec{e}_y + z\vec{e}_z = \begin{pmatrix} x' \\ y' \end{pmatrix}$$

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} xr_x\cos(\varphi_x) + yr_y\cos(\varphi_y) + zr_z\cos(\varphi_z) \\ xr_x\sin(\varphi_x) + yr_y\sin(\varphi_y) + zr_z\sin(\varphi_z) \end{pmatrix}$$

3.3 Computer implementations of the matrix and the transformation

3.3.1 Generic computer code

The following is example code for various computer systems.

```
x_ = x*r*cos(alpha) + y*r*cos(beta) + z*r*cos(gamma)
y_ = x*r*sin(alpha) + y*r*sin(beta) + z*r*sin(gamma)
```

3.3.2 JavaScript computer code

This is a full EcmaScript 6 snippet with all neccessary informations.

```
let rad = (deg) => Math.PI/180*deg;
let r_x = 1, r_y = 1, r_z = 1;
let phi_x = rad(220), phi_y = rad(330), phi_z = rad(90);
let xAxisCos = r_x*Math.cos(phi_x),
    yAxisCos = r_y*Math.cos(phi_y),
    zAxisCos = r_z*Math.cos(phi_z),
    xAxisSin = r_x*Math.sin(phi_x),
    yAxisSin = r_y*Math.sin(phi_y),
    zAxisSin = r_z*Math.sin(phi_z);
let transform2d = ([x,y,z]) => [
    x*xAxisCos+ y*yAxisCos+ z*zAxisCos,
    x*xAxisSin+ y*yAxisSin+ z*zAxisSin];
let transform2dAll = (P) => P.map(transform2d);

let examplePoints = transform2dAll([[1,2,3], [3,4,5], [14,24,15]]);
```

Remark I already proved affine transformations (Rotation, Scaling, Translation), with and without applying a local 3x3 base to the object (with is better, let $r_x = r_y = r_z$), then projecting the points afterwards with this formula. I think the alternative graphics algorithm up to there could find a place here.

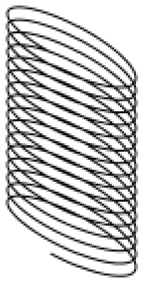

Figure 4: A helix shown as (x,y,z)=f(t) with implement.html on a Canvas2DRenderingContext testing the javascript example code.

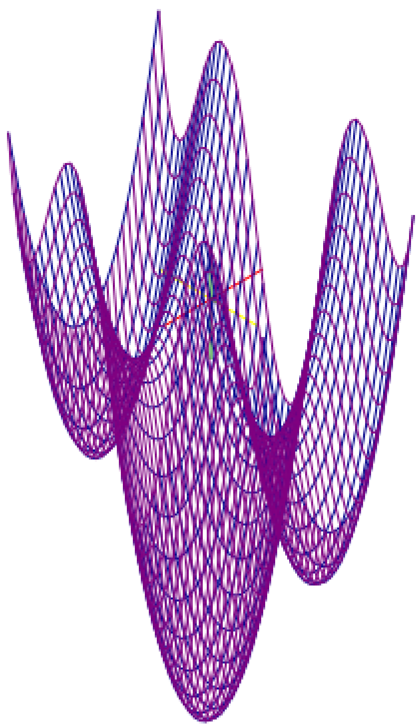

Figure 5: $x^2 + y^2 + 3y \sin y$ from [-5,5] and [-3,3] as z=f(x,y) with cheap3danimate.html on a Canvas2DRenderingContext

4 Corollary

4.1 Converting four Dimensions down to two dimensions

The theorem can be used to handle more dimensions, for example can four two-dimensional vectors represent a 4-D space on the 2-D plane. They get converted into the correct 2-D points. For Example, if you use a 2x4 matrix and convert all points at each instance of t you have a moving object into the direction of the fourth basis vector.

$$\boldsymbol{A} := \begin{pmatrix} \vec{e}_x & \vec{e}_y & \vec{e}_z & \vec{e}_t \end{pmatrix} = \begin{pmatrix} r_x\cos(\varphi_x) & r_y\cos(\varphi_y) & r_z\cos(\varphi_z) & r_t\cos(\varphi_t) \\ r_x\sin(\varphi_x) & r_y\sin(\varphi_y) & r_z\sin(\varphi_z) & r_t\sin(\varphi_t) \end{pmatrix}$$

Here the basis is four times of two dimensions. A 2x4 matrix with four two dimensional basis vectors, one for each axis.

$$\boldsymbol{A}\begin{pmatrix} x \\ y \\ z \\ t \end{pmatrix} = \sum_n \vec{e}_n\vec{x}_n = \begin{pmatrix} x' \\ y' \end{pmatrix}$$

Proof:

$$\boldsymbol{A}\begin{pmatrix} x \\ y \\ z \\ t \end{pmatrix} = \begin{pmatrix} xr_x\cos(\varphi_x) + yr_y\cos(\varphi_y) + zr_z\cos(\varphi_z) + zr_t\cos(\varphi_t) \\ xr_x\sin(\varphi_x) + yr_y\sin(\varphi_y) + zr_z\sin(\varphi_z) + zr_t\sin(\varphi_t) \end{pmatrix}$$

$$= x\vec{e}_x + y\vec{e}_y + z\vec{e}_z + t\vec{e}_t = \sum_n \vec{e}_n\vec{x}_n = \begin{pmatrix} x' \\ y' \end{pmatrix}$$

The same method can be used to convert any other number of dimensions to the xy-plane. But it can also be used in a generic m by n case[5], to convert from n dimensions down to m, if you know the basis for the destination.

5 Summary

5.1 Summary of all neccessary steps

1. Lay out the three basis vectors around a circle and write down the angles φ_n. Programmers have to write down a variable for anyways.
2. Write down the basis vectors $\vec{e}_n$ as $r_n\cos\varphi_n$ and $r_n\sin\varphi_n$ (two dimensional). Dont multiply with r_n for a unit length of 1 or multiply with r_n to change the length of the basis vector.
3. Put the three basis vectors $\vec{e}_n$ into a matrix A. Programmers can directly code the two lines of multiplication and forget the formal rest.
4. Iterate over your points and multiply each (x, y, z) with the matrix $\boldsymbol{A}$, which acts as a linear operator, and put (x', y') into your new set.

Remark

About the word *unit*. I am not really sure, if i have to use *base vector* for a vector of any length and *unit vector* only for the *unit length* of 1. Because of the misleading mismatch with the *unit* of the thought *coordinate axes*, which the *base vector* defines, i tend in the first versions to misuse the word *unit vector* for both. If you find this, or any other formal mistake, be sure, it is not wanted :-) I will try to remove more of these spelling errors[6] in the next versions.

[5] http://de.wikipedia.org/wiki/Abbildungsmatrix, also found in lecture scripts, but not anyone explaining me this matrix here or the topic of the from-three-to-two-points conversion. Is it too obvious? Or isnt it obvious?

[6] The *Gerholdian operator*, the *Gerholdian basis*, the *Gerhold projection matrix*, the *Gerhold transformation* are my favourite nicknames for my late discovery, making sure, the three two dimensional and trigonometric basis vectors, which i explained, sit in the matrix.

A Proving more rules of vector spaces

A.1 The origin stays in the origin

A trivial proof is to prove, that the zero vector $\vec{0} \in R^3$ maps to the zero vector $\vec{0} \in R^2$.

Proof:

$$\boldsymbol{A}\begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 0+0+0 \\ 0+0+0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

A.2 Points along one axis

Another trivial proof is to prove, that coordinates lying on one axis are a multiple of the basis vector of the axis.

Proof:

$$\boldsymbol{A}\begin{pmatrix} a \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} ar_x \cos\varphi_x + 0 + 0 \\ ar_x \sin\varphi_x + 0 + 0 \end{pmatrix} = a\vec{e}_x$$

$$\boldsymbol{A}\begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 + r_y \cos\varphi_y + 0 \\ 0 + r_y \sin\varphi_y + 0 \end{pmatrix} = \vec{e}_y$$

$$\boldsymbol{A}\begin{pmatrix} 0 \\ 0 \\ -b \end{pmatrix} = \begin{pmatrix} 0 + 0 - br_z \cos\varphi_z \\ 0 + 0 - br_z \sin\varphi_z \end{pmatrix} = -b\vec{e}_z$$

A.3 Multiplications with constants

Another trivial proof is to show, that $\boldsymbol{A}(\lambda\vec{x}) = \lambda\boldsymbol{A}\vec{x}$. It doesnt matter, where you multiply with the constant. You can multiply the original vector, or the resulting vector. You reach the same point.

Proof:

$$\begin{aligned}
\boldsymbol{A}(\lambda\vec{x}) &= \boldsymbol{A}\begin{pmatrix} \lambda x \\ \lambda y \\ \lambda z \end{pmatrix} \\
&= \begin{pmatrix} \lambda x r_x \cos(\varphi_x) + \lambda y r_y \cos(\varphi_y) + \lambda z r_z \cos(\varphi_z) \\ \lambda x r_x \sin(\varphi_x) + \lambda y r_y \sin(\varphi_y) + \lambda z r_z \sin(\varphi_z) \end{pmatrix} \\
&= \lambda \begin{pmatrix} x r_x \cos(\varphi_x) + y r_y \cos(\varphi_y) + z r_z \cos(\varphi_z) \\ x r_x \sin(\varphi_x) + y r_y \sin(\varphi_y) + z r_z \sin(\varphi_z) \end{pmatrix} \\
&= \lambda \begin{pmatrix} x' \\ y' \end{pmatrix} \\
&= \lambda\boldsymbol{A}\vec{x}
\end{aligned}$$

A.4 Additions and subtractions

Another trivial proof is to show, that $\boldsymbol{A}(\vec{v}+\vec{w}) = \boldsymbol{A}\vec{v} + \boldsymbol{A}\vec{w}$. It does not matter, if you add the original or the results . The outcome is the same point, the same vector.

Proof:

$$\begin{aligned}
\boldsymbol{A}\begin{pmatrix} x+u \\ y+v \\ z+w \end{pmatrix} &= \begin{pmatrix} (x+u)r_x\cos(\varphi_x) + (y+v)r_y\cos(\varphi_y) + (z+w)r_z\cos(\varphi_z) \\ (x+u)r_x\sin(\varphi_x) + (y+v)r_y\sin(\varphi_y) + (z+w)r_z\sin(\varphi_z) \end{pmatrix} \\
&= \begin{pmatrix} xr_x\cos(\varphi_x) + yr_y\cos(\varphi_y) + zr_z\cos(\varphi_z) \\ xr_x\sin(\varphi_x) + yr_y\sin(\varphi_y) + zr_z\sin(\varphi_z) \end{pmatrix} + \begin{pmatrix} ur_x\cos(\varphi_x) + vr_y\cos(\varphi_y) + wr_z\cos(\varphi_z) \\ ur_x\sin(\varphi_x) + vr_y\sin(\varphi_y) + wr_z\sin(\varphi_z) \end{pmatrix} \\
&= \begin{pmatrix} x' \\ y' \end{pmatrix} + \begin{pmatrix} u' \\ v' \end{pmatrix} \\
&= \boldsymbol{A}\begin{pmatrix} x \\ y \\ z \end{pmatrix} + \boldsymbol{A}\begin{pmatrix} u \\ v \\ w \end{pmatrix}
\end{aligned}$$

A.5 Rule of linearity

Corollary From the previous two proofs, it is obvious to see, that

$$\boldsymbol{A}(\lambda\vec{v} + \kappa\vec{w}) = \lambda\boldsymbol{A}\vec{v} + \kappa\boldsymbol{A}\vec{w} = \lambda\begin{pmatrix} x' \\ y' \end{pmatrix} + \kappa\begin{pmatrix} u' \\ v' \end{pmatrix}$$

which is a standard formulation of the rule of linearity. For example, you can find this rule in the form $\boldsymbol{A}(c\vec{x} + d\vec{y}) = c\boldsymbol{A}\vec{x} + d\boldsymbol{A}\vec{y}$ in [3], but also in every linear algebra 1 lecture script.

A.6 About the norm

The norm used is the euclidean norm, or the 2-norm. This is the square root of the sum of the squares of the absolute values of the components $\|\vec{x}\| = \sqrt{\sum_{i=1}^{n} |x_n|^2}$. In linear algebra, functional analysis and topology lectures there are three fundamental properties of the norm. Definiteness, homogenity and the triangle inequality.

Definitness Show that $\|\vec{x}\| = 0$ if $\vec{x} = 0$

$$\|\vec{x}\| = \|\vec{0}\| = \sqrt{0^2 + 0^2} = 0$$

Homogenity Show that $\|a\vec{x}\| = |a|\|\vec{x}\|$

$$\|a\vec{x}\| = \sqrt{|a\vec{x}_1|^2 + |a\vec{x}_2|^2} = \sqrt{|a|^2(|\vec{x}_1|^2 + |\vec{x}_2|^2)} = |a|\sqrt{|\vec{x}_1|^2 + |\vec{x}_2|^2} = |a|\|\vec{x}\|$$

Triangle inequality (Minkowski inequality) Show that $\|\boldsymbol{A}(\vec{v}+\vec{w})\| \leq \|\boldsymbol{A}\vec{v}\| + \|\boldsymbol{A}\vec{w}\|$

$$\sqrt{\sum_{i=1}^{n} |\vec{v}_i + \vec{w}_i|^2} \leq \sqrt{\sum_{i=1}^{n} |\vec{v}_i|^2} + \sqrt{\sum_{i=1}^{n} |\vec{w}_i|^2}$$

Remark This subsection is not complete.

A.7 Metrics

Where a norm is, there will be a metric induced. The measurement of the distance between two points is defined by the d-function. It is the length of the difference vector between the two points.

$$d(\vec{x},\vec{y}) = \|\vec{x}-\vec{y}\| = \sqrt{\sum_{i=1}^{n} |\vec{x}_i - \vec{y}_i|^2}$$

Metrics have three fundamental properties.

1. If the distance is zero, the vectors are equal.
$$d(x,y) = 0 \iff x = y$$
2. It does not matter, whether you read $d(x,y)$ or $d(y,x)$, the number must be equal.
$$d(x,y) = d(y,x)$$
3. The third one is the triangle inequality. Going over another point is always a step longer.
$$d(x,z) \leq d(x,y) + d(y,z)$$

Remark This subsection is not complete and has to be continued. The point is to measure now the difference between the original coordinates and the new planar coordinates. $d(\vec{x},\vec{y})_2 = \|\vec{x}-\vec{y}\|_2 \leq d(\vec{x},\vec{y})_3 = \|\vec{x}-\vec{y}\|_3$ is my assumption which is not proven now.

A.8 Transpose and unproven

Remark Less trivial is to figure out, what the transpose and the products with the transpose and the inverse matrices of those products are good for. Our rectangular 2x3 basis matrix has a transponse. A 3x2 matrix. The products are $\boldsymbol{A}\boldsymbol{A}^T$, a 2 by 2 matrix, and $\boldsymbol{A}^T\boldsymbol{A}$, a 3 by 3 matrix. Today i will just show the transpose.

$$\begin{pmatrix} r_x\cos(\varphi_x) & r_y\cos(\varphi_y) & r_z\cos(\varphi_z) \\ r_x\sin(\varphi_x) & r_y\sin(\varphi_y) & r_z\sin(\varphi_z) \end{pmatrix}^T = \begin{pmatrix} r_x\cos(\varphi_x) & r_x\sin(\varphi_x) \\ r_y\cos(\varphi_y) & r_y\sin(\varphi_y) \\ r_z\cos(\varphi_z) & r_z\sin(\varphi_z) \end{pmatrix}$$

Multiplying out the transposes yield the following forms.

$$\boldsymbol{A}\boldsymbol{A}^T = \begin{pmatrix} \sum_{i=1}^3 r_n\cos\varphi_n & \sum_{i=1}^3 r_n^2\cos\varphi_n\sin\varphi_n \\ \sum_{i=1}^3 r_n^2\cos\varphi_n\sin\varphi_n & \sum_{i=1}^3 r_n\sin\varphi_n \end{pmatrix} = \begin{pmatrix} a & b \\ b & c \end{pmatrix}$$

You see, in the 2x2 matrix $\boldsymbol{A}\boldsymbol{A}^T$ is $a_{ij} = a_{ji}$. In the 3x3 matrix $\boldsymbol{A}^T\boldsymbol{A}$ is also $a_{ij} = a_{ji}$. I will abbreviate $\cos\varphi_n$ with C_n and $\sin\varphi_n$ with S_n.

Remark I have not compared with the sin(A+B) formulas.

$$\boldsymbol{A}^T\boldsymbol{A} = \begin{pmatrix} C_x^2+S_x^2 & C_xC_y+S_xS_y & C_xC_z+S_xS_z \\ C_yC_x+S_yS_x & C_y^2+S_y^2 & C_yC_z+S_yS_z \\ C_zC_x+S_zS_x & C_zC_y+S_zS_y & C_z^2+S_z^2 \end{pmatrix} = \begin{pmatrix} r_x^2 & a & b \\ a & r_y^2 & c \\ b & c & r_z^2 \end{pmatrix}$$

Remark I wrote down the formula for the 2x2 determinant and noticed, its getting hairy writing with bare hands. Using numeric software like FreeMat should be less stressing. Whether it makes sense to calculate the determinants and the inverses or not, can not be told from looking at these matrices by a guy like me. This topic is continued in the next versions.

Remark Missing are $|\boldsymbol{A}\boldsymbol{A}^T|$ and $|\boldsymbol{A}^T\boldsymbol{A}|$ and $(\boldsymbol{A}\boldsymbol{A}^T)^{-1}$ and $(\boldsymbol{A}^T\boldsymbol{A})^{-1}$ and various tries to combine them to P, to x^TAx, and and.

References

Michael Corral, Schoolcraft College, Vector Calculus, GNU Free Documentation License, http://mecmath.net

Michael Corral, Schoolcraft College, Trigonometry, GNU Free Documentation License, http://mecmath.net

Gilbert Strang, MIT, Linear Algebra and its Applications. Fourth Edition.

Gilbert Strang, MIT, Calculus. MIT OpenCourseWare Supplemental Resources. http://ocw.mit.edu

Michael Corral, Schoolcraft College, Latex Mini Tutorial, http://mecmath.net

Manuela Jürgens, Thomas Feuerstack, Fernuniversität Hagen, LaTeX, eine Einführung und ein bisschen mehr..., a026_latex_einf.pdf

Dr.Jan Rudl, Technische Universität Dresden, Fachbereich Mathematik, Einführung in LaTeX, LaTeX-Kurs.pdf