

# 1 Introduction

## 1.1 Background

R can provide more statistical graph. The core graphics system can be divided into two main packages. The first package is the graphics package. It is older and it provides the original GRZ graphics system from S. It is fast and it can provide traditional graphics. The newer package is the grid package. It is actually slower but it has more power on flexibility and additional features than the graphics package.

The graph is drawn by using grid can be editing in many more ways than the graph been drawn on the basic graphics package. However, there is a new package, called gridGraphics, which allows us to convert the plot been draw by graphics package to grid graphics. So that the plot can be manipulated by any function on the grid package in order to provide more statistical plot.

The main function on the gridGraphics is called `grid.echo()`, which allows any plot been drawn in the current graphics device convert to the grid graphics. The following codes provide a quick example of the demonstration. We generate 25 random number of x and y. First, we draw a scatter plot by using the function `plot()` from the basic graphic package, then we redraw it by using the `grid.echo()` from the gridGraphic package with grid.

```
> x = runif(25)
> y = runif(25)
> plot(x,y, pch = 16)
> grid.echo()
```

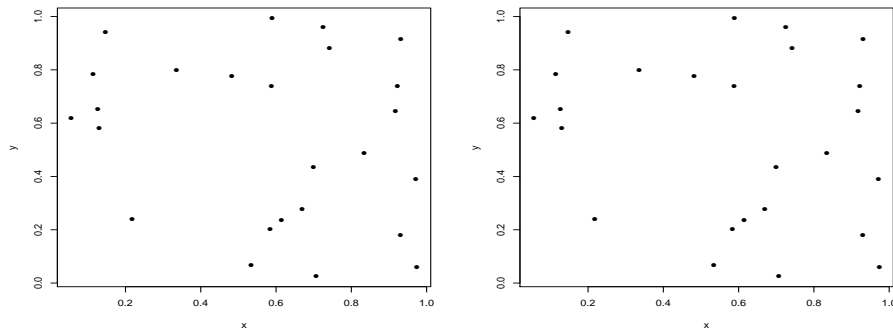


Figure 1: The left plot drawn by using `plot()`, the Right plot is redraw it by using `grid.echo()` on grid graphic system, overall, they are identical to each other

## 1.2 The problem

The `grid.echo()` can replicate most of plot that drawn by the graphics package. However, there are few functions on the graphics package that cannot replicate. One will the the `persp()` which for drawing 3-dimemntional surfaces on the basic graphics.

```
> x = seq(-10,10,length = 100)
> y = seq(-10,10,length = 100)
> f <- function(x, y) { r <- sqrt(x^2+y^2); 10 * sin(r)/r }
> z <- outer(x, y, f)
> z[is.na(z)] <- 1
> persp(x, y, z, theta = 20, phi = 30, expand = 0.5, box = TRUE)
> grid.echo()
warning message:
In FUN(x[[i]], ...) : gridGraphics cannot emulate persp()
```

Figure 2: an example shows that the `grid.echo()` cannot replicate the `persp()` on grid

It leads to our aim of this project. Firstable, it is necessary to understand some important theory behind the 3-dimentional plot, such as the transformation from 3-D to 2-D, the drawing order of each ploygons (will be explained later). ect. Then we try to redraw it by using the grid package and we can replicate this function on grid by using the `grid.echo()`.

## 1.3 Aim of the project

This project has following aims: 1. understanding the details of theory of the transformation from 3-D to 2-D, In order to do that, the fast way is go through and understand the C code that already build in the graphic package. 2. After we have some knowledge about the theory, then It is a good idea that translate the C code to R code as simliar as possible. 3. rewrite the R code in a more efficiency way.

# 2 Methodology

## 2.1 standalone

Drawing a 3-dimenssion object on a 2-dimenssion plane requires a transformation. that is, the 3-D transformation matrix. However, the `persp()` will return the transformation matrix.

```
> x = y = seq(-10,10,length = 60)
> f = function(x, y) { r <- sqrt(x^2+y^2); 10 * sin(r)/r }
> z = outer(x, y, f); z[is.na(z)] <- 1
> trans <- persp(x, y, z, theta=120, phi = 20, expand = 0.5)
> trans
```

	[,1]	[,2]	[,3]	[,4]
[1,]	-5.000000e-02	-0.02961981	0.08137977	-0.08137977
[2,]	8.660254e-02	-0.01710101	0.04698463	-0.04698463
[3,]	-4.390837e-18	0.07781011	0.02832056	-0.02832056
[4,]	1.697562e-17	-0.30082519	-2.84154222	3.84154222