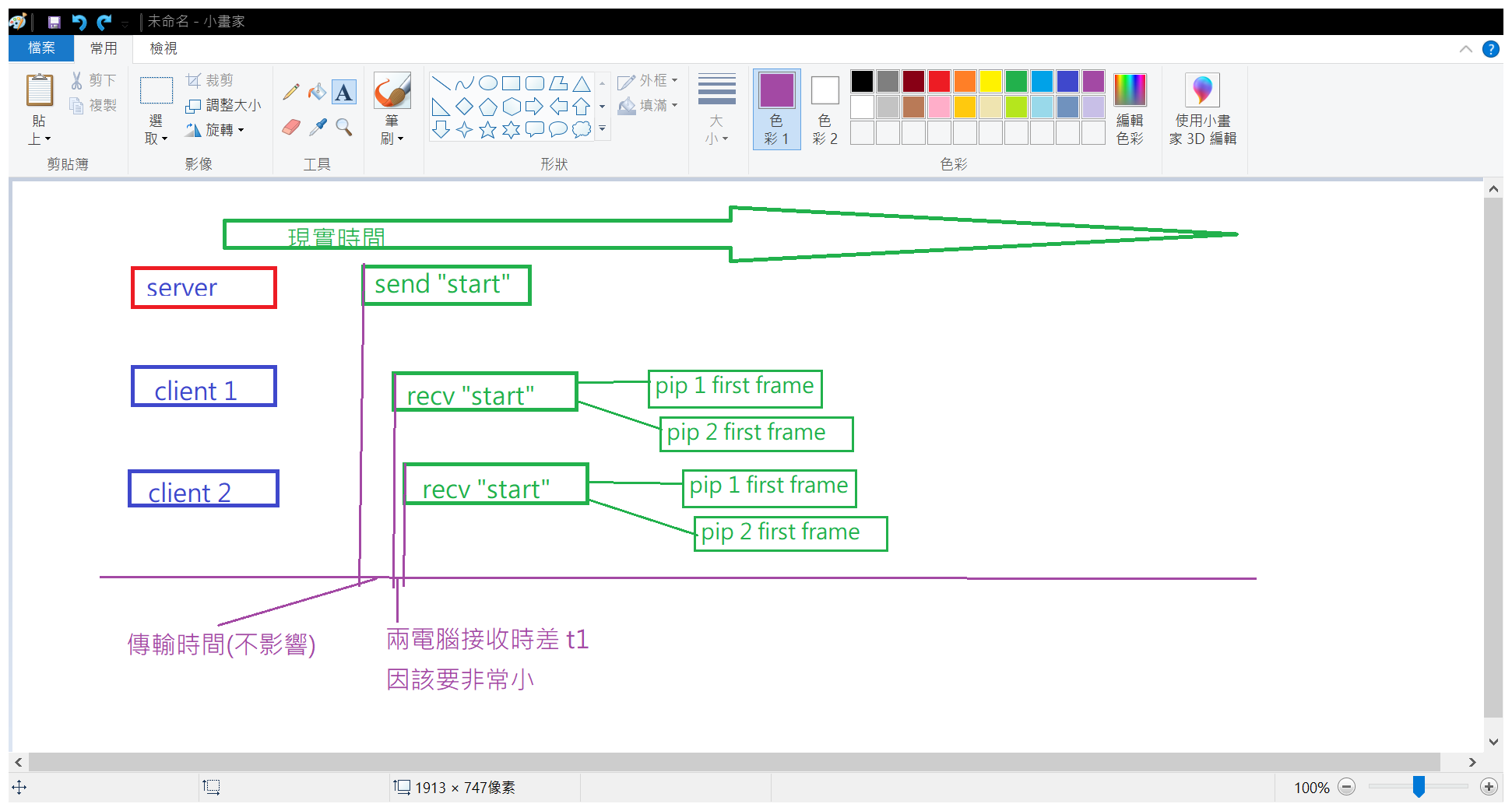
實驗大綱

這次實驗的目的是確認多台realsense 的frames 有沒有在時間上對齊

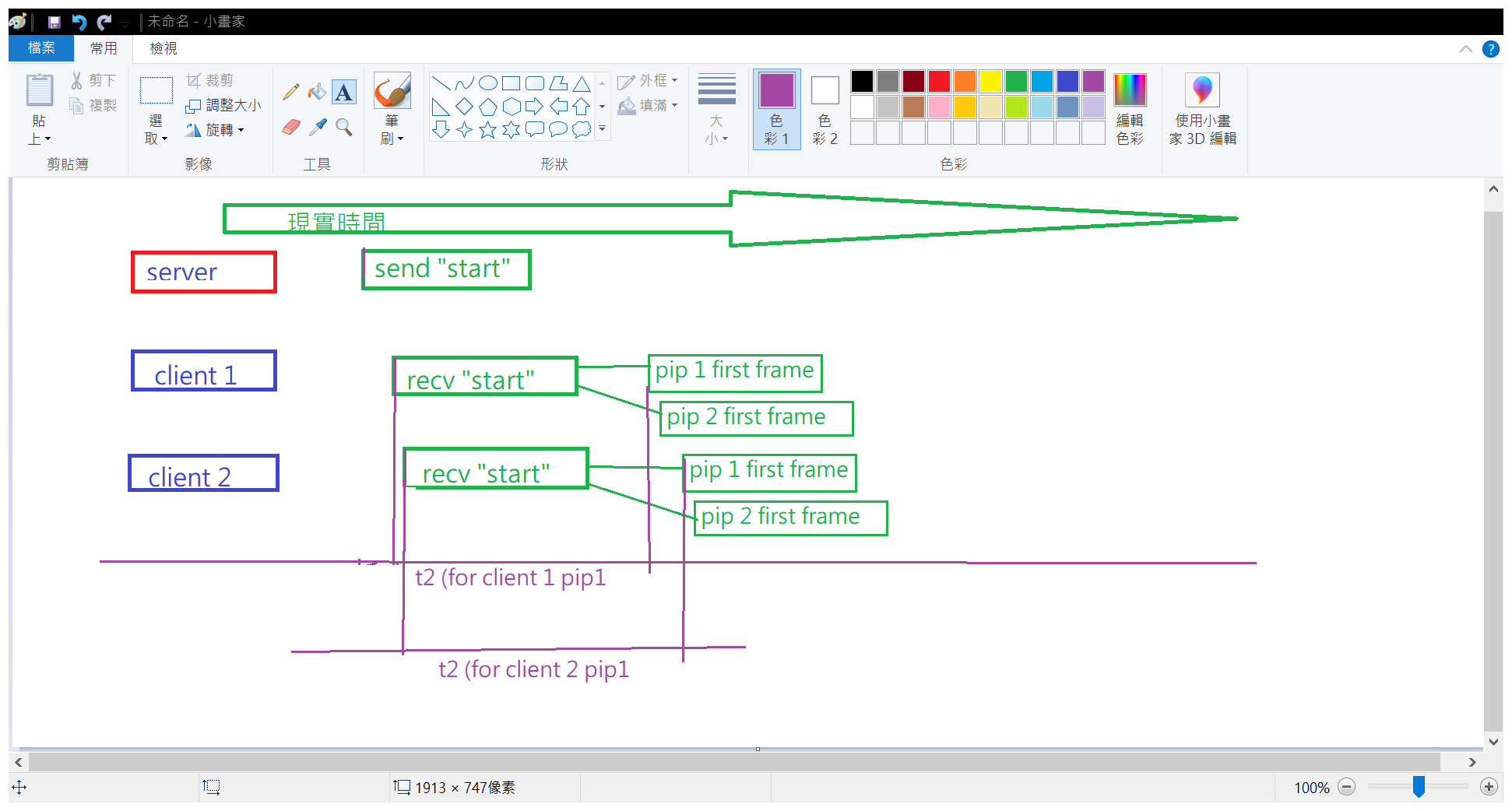
示意圖

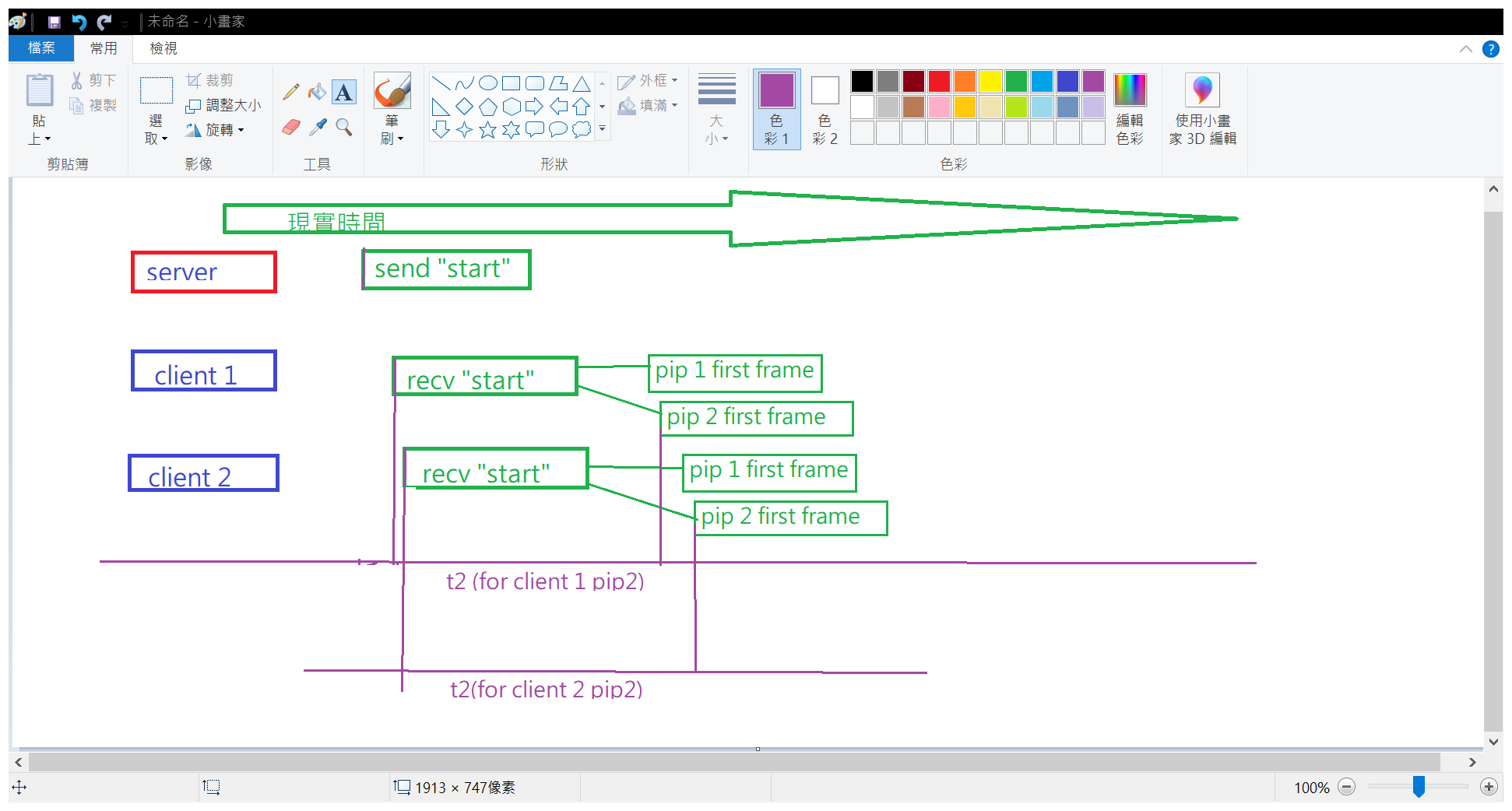
第一個誤差t1 ，這是不同電腦接收同一訊號可能依然有一點誤差

(可能來自 作業系統，網卡 的各種狀況)



t2 從接收到指令到錄製第一frame 所需時間(可以直接由程式自動量測)





因此假設我們手上的兩個bag 檔(假如是來自client 1 pip1, client 2 pip2)

我們想要知道兩個bag 檔的 第一frame 的現實時間是否match ，

還有如果不match ，要如何教準。

我們假設client 1 pip1 bag frame1 timestamp = t3

client 2 pip2 bag frame1 timestamp = t4

並且每個bag 裡都拍到同一個led 亮起瞬間

假設 client 1 pip1 bag frame led timestamp = t5

client 2 pip2 bag frame led timestamp = t6

我們可以保證t5 t6發生在同一個現實時間

所以我們重新將t5 t6 設定為兩bag時間軸的 第 0 秒

因此

client 1 pip1 bag frame led timestamp = 0

client 2 pip2 bag frame led timestamp = 0

client 1 pip1 bag frame1 timestamp = t3-t5

client 2 pip2 bag frame1 timestamp = t4-t6

我們可觀察以上兩者是否相同，如果有一些誤差表示其實兩bag frame1不對齊

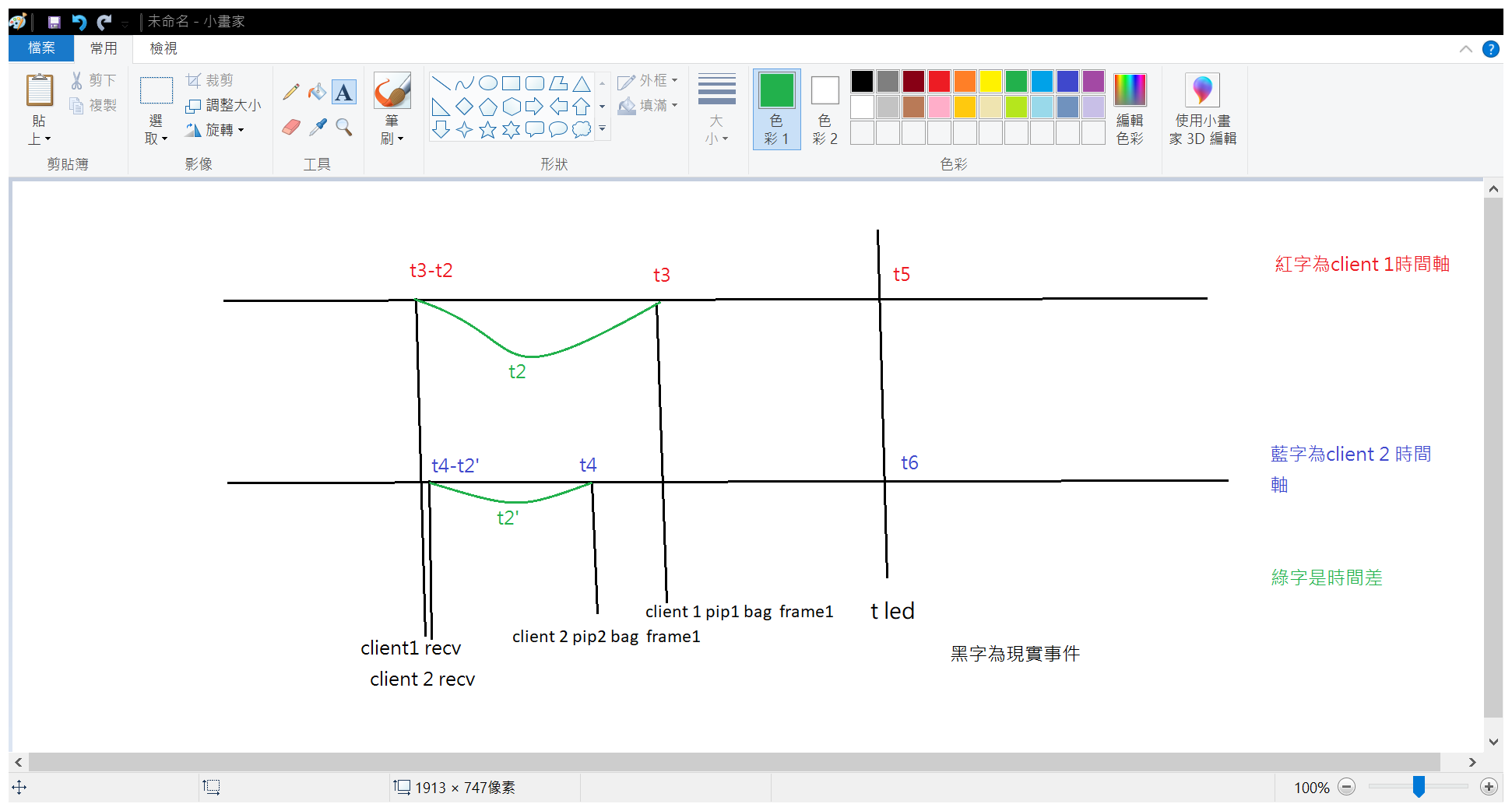
也就是兩bag frame 1 不是在同一現實時間錄製

所以或許我們需要一些校準方式。

我們已知(由程式量出)(上一頁定義)

t2 (for client 1 pip1) 🡺t2 簡稱

t2 (for client 2 pip2) 🡺t2’



此時 t3-t5-t2= t client1接收訊號

t4-t6-t2’= t client2接收訊號

如果t client1接收訊號 almost == t client2接收訊號

也就是兩者收到訊號的現實時間幾乎相同

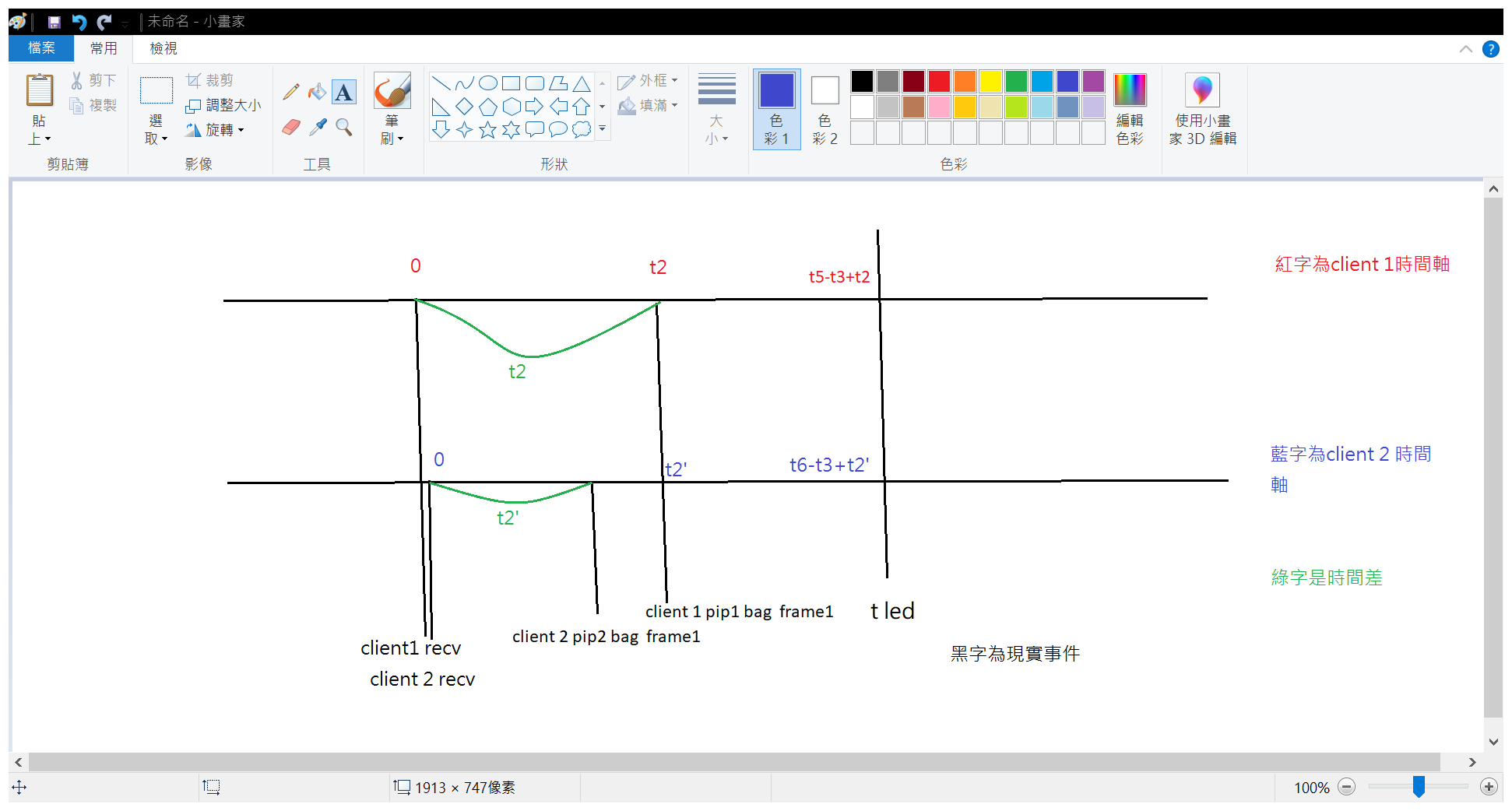
表示我們可以用將收到訊號的時間視為第零秒

client 1 pip1 bag frame1 =t2

client 2 pip2 bag frame1 =t2’

client 1 pip1 bag frame led timestamp =t5-t3+t2

client 2 pip2 bag frame led timestamp = t6-t4+t2’



將來只要將所有client 1 pip1 bag 的tn –t3 +t2 (tn 是任一frame timestamp)

client 2 pip2 bag 的 tm –t4+t2’

就可以校準好了了

實際實驗流程

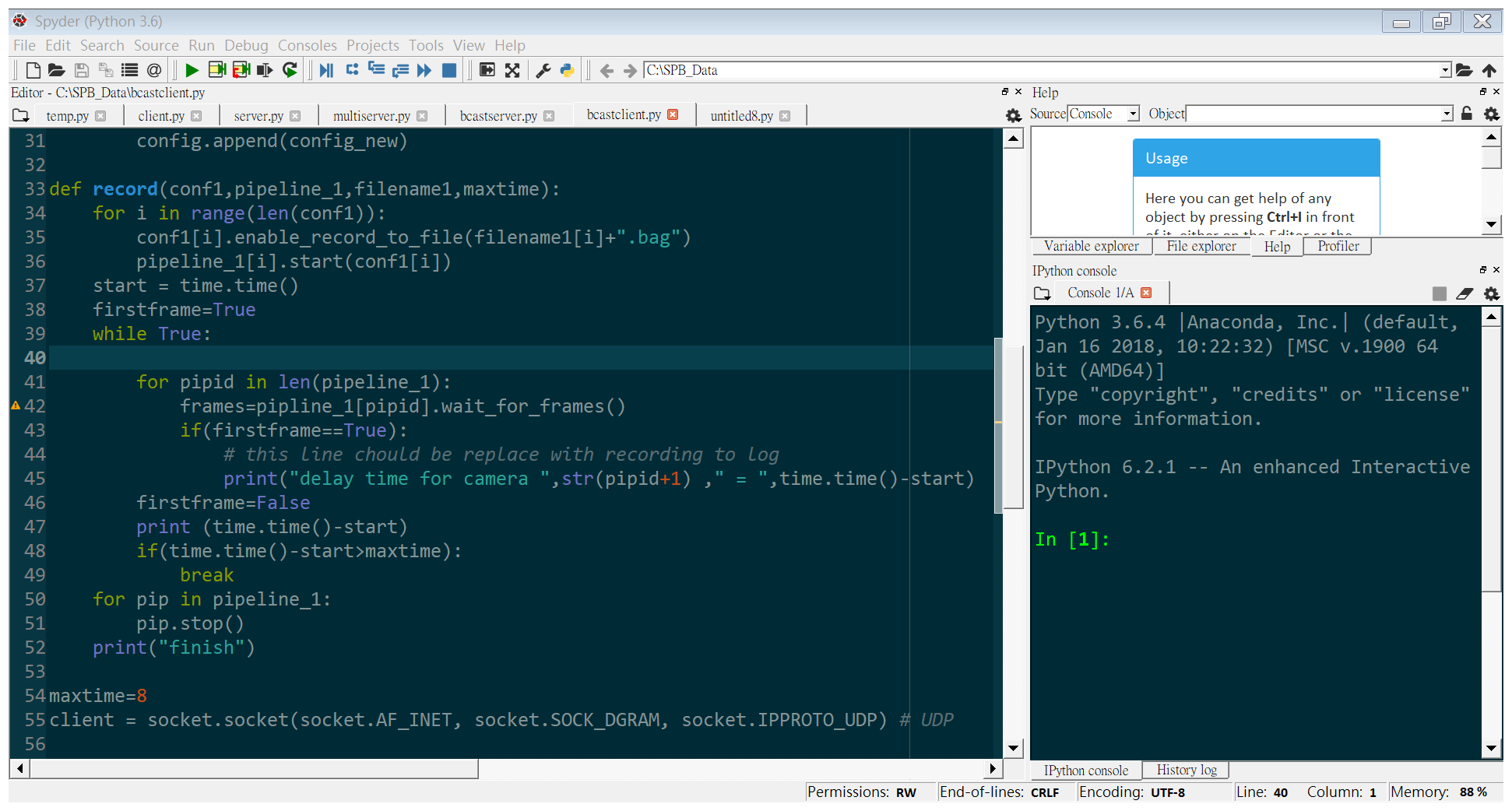
準備多台realsense 使用同步錄製錄好led 開燈

抓出每個bag 的 first frame timestamp , led frames timestamp

還有錄製每個bag 前紀錄的t2

計算 t led –tfirst ,t led –tfirst +t2

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | t2 | t first frame | t led | t led –  tfirst frame | t led+t2  -tfirst frame |
| bag 1 |  |  |  |  |  |
| bag2 |  |  |  |  |  |
| bag 3 |  |  |  |  |  |
| bag4 |  |  |  |  |  |
| bag5 |  |  |  |  |  |
| bag6 |  |  |  |  |  |



t2 會在這個修改過後的這一行

# -\*- coding: utf-8 -\*-

"""

Created on Wed Jul 8 09:53:36 2020

@author: gordon

"""

import socket

import pyrealsense2 as rs

import time

realsense\_ctx = rs.context()

pipeline\_1 =[]

config\_1=[]

connected\_devices = []

for i in range(len(realsense\_ctx.devices)):

detected\_camera = realsense\_ctx.devices[i].get\_info(rs.camera\_info.serial\_number)

connected\_devices.append(detected\_camera)

print("connected device : ",connected\_devices)

for i in connected\_devices:

config\_new = rs.config()

config\_new.enable\_device(i)

config\_new.enable\_stream(rs.stream.depth, 1280, 720, rs.format.z16, 30)

config\_new.enable\_stream(rs.stream.color, 1280, 720, rs.format.rgb8, 30)

config\_1.append(config\_new)

pipeline\_1.append(rs.pipeline())

def initconfig(connected\_devices,config):

for i in connected\_devices:

config\_new = rs.config()

config\_new.enable\_device(i)

config\_new.enable\_stream(rs.stream.depth, 1280, 720, rs.format.z16, 30)

config\_new.enable\_stream(rs.stream.infrared, 1280, 720, rs.format.y8, 30)

config.append(config\_new)

def record(conf1,pipeline\_1,filename1,maxtime):

start = time.time()

for i in range(len(conf1)):

conf1[i].enable\_record\_to\_file(filename1[i]+".bag")

pipeline\_1[i].start(conf1[i])

firstframe=True

while True:

for pipid in len(pipeline\_1):

frames=pipline\_1[pipid].wait\_for\_frames()

if(firstframe==True):

# this line chould be replace with recording to log

print("delay time for camera ",str(pipid+1) ," = ",time.time()-start)

firstframe=False

print (time.time()-start)

if(time.time()-start>maxtime):

break

for pip in pipeline\_1:

pip.stop()

print("finish")

maxtime=8

client = socket.socket(socket.AF\_INET, socket.SOCK\_DGRAM, socket.IPPROTO\_UDP) # UDP

# Enable port reusage so we will be able to run multiple clients and servers on single (host, port).

# Do not use socket.SO\_REUSEADDR except you using linux(kernel<3.9): goto https://stackoverflow.com/questions/14388706/how-do-so-reuseaddr-and-so-reuseport-differ for more information.

# For linux hosts all sockets that want to share the same address and port combination must belong to processes that share the same effective user ID!

# So, on linux(kernel>=3.9) you have to run multiple servers and clients under one user to share the same (host, port).

# Thanks to @stevenreddie

client.setsockopt(socket.SOL\_SOCKET, socket.SO\_REUSEADDR, 1)

# Enable broadcasting mode

client.setsockopt(socket.SOL\_SOCKET, socket.SO\_BROADCAST, 1)

client.bind(("", 37020))

recvstring=""

beforeStr=""

while True:

print("waiting for commend to start recording...")

# Thanks @seym45 for a fix

data, addr = client.recvfrom(1024)

beforeStr=recvstring

recvstring=data.decode("utf-8")

if(beforeStr==recvstring):

print("should not use same filename...")

continue

if(recvstring=="q"):

print("end client section ,goodbye")

break

spl=recvstring.split()

if(spl[0]=="set"):

maxtime=int(spl[1])

print("set max recording time to : ",maxtime)

continue

print("received message: ",recvstring)

record(config\_1,pipeline\_1,[recvstring+"c1",recvstring+"c2"],maxtime)