Sdc\_hw4

311512064鄧書桓

1. Introduction

Kalman filter 是一種很厲害的濾波器，它能夠通過我們擁有已知的訊息(觀察得到的數據或過去的狀態)與belief的分佈相結合來減輕現實生活中的不確定性，並利用誤差與noise來跟蹤我們所要求的目標或測量。簡單來說，就是通過現實生活中感測器的誤差與環境中的不確定性，不斷的迭待更新belief與預測，來降低不確定性，進而提高準確率。

1. Briefly explain your code.

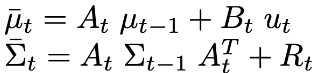
將卡爾曼濾波器的更新修正方式程式化。

一張含有 文字 的圖片

自動產生的描述

首先，將Q與R設置成所想要的矩陣，如何求得與詳細過程會在下面第四點做說明。

接著，根據以下公式來完成預測:



相對應的code:

一張含有 文字 的圖片

自動產生的描述

利用numpy中的dot()函式來實現，其中x為算式中的 一張含有 文字 的圖片

自動產生的描述 、P為 ，值得注意的是，這邊定義的Q與R與上課教的相反(此程式的Q為公式中的R)。

最後，透過以下公式來完成更新: 一張含有 文字 的圖片

自動產生的描述

相對應的code:

一張含有 文字 的圖片

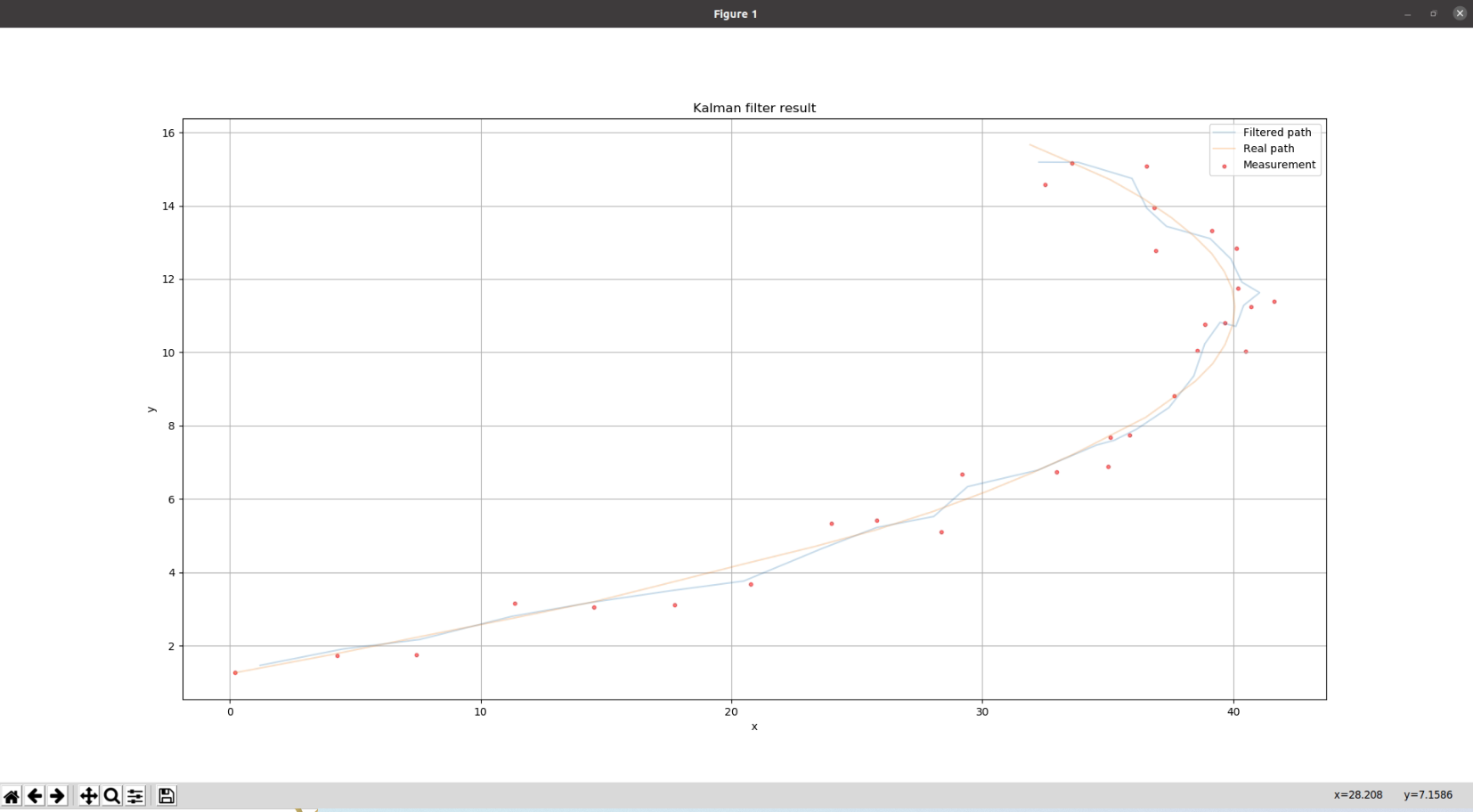
自動產生的描述

這邊同樣使用numpy.dot()實現，H為算是中的C 、x為 一張含有 文字 的圖片

自動產生的描述 、P為 ，由於怕式子太長，有將部分括號內的算式以新的變數命名，如第一行的y(代表 與第二行的s(代表 一張含有 文字 的圖片

自動產生的描述)，最終返回x與p方便做下一次的更新與預測。

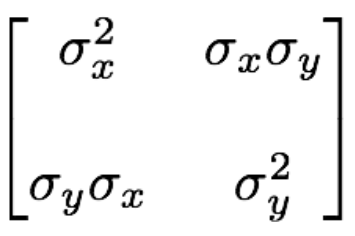
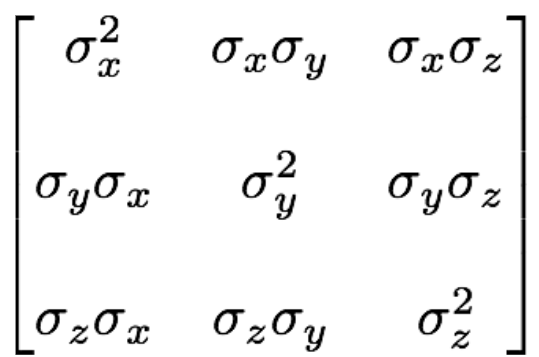
1. Include the output image.



由此圖可知，根據所調整的Q與R能使卡爾曼濾波器製造的路徑與實境的路徑幾乎一樣，但仍有些許的誤差，若想要做到幾乎一樣，我認為不能給定固定的Q與R，需要在每次迭代時一起做更新調整。

1. How you design the covariance matrices(Q, R)?

Q與R先由以下公式推出初始值:

其中，因為x、y與yaw相互獨立，因此Q與R為對角矩陣，又根據題目所給的標準差可得:

Q = [ [0.1, 0.0, 0.0], R = [ [0.75, 0.0],

[0.0, 0.1, 0.0], [0.0, 0.75] ]

[0.0, 0.0, 0.1] ]

但是，設成這樣filter的誤差很大，預測出來的路徑與real path差很多，因此還需做一些調整。其中，提高Q的值(上課教的R)可使filter更不相信自己的預測，也就是更相信觀測到的數據;提高R的值(上課教的Q) 可使filter更不相信自己的觀測，也就是更相信預測的數據。藉由此觀念來進行try and err，最終調整成第二點的數據。

1. How will the value of Q and R affect the output of Kalman filter?

影響的方式在第四點有明確說明，簡單來說，就是他們的值會影響filter比較相信預測的數據還是觀測到的數據，我也是根據此原理來調整Q與R。