

Computer Vision: from Recognition to Geometry

期末專題實作報告

R06942114 劉彥廷 R07942063 郭冠軒 R07942074 李元顥 R07942077 楊福恩

影像前處理 (Image Preprocessing)

1. 直方圖均衡化

由於本次專題提供之部分測資有左右眼之影像亮度及色調不均之現象，因此在進入視差計算前，我們先將兩張輸入圖進行直方圖均衡化，亦即調整兩張影像之對比度直至相近或相等為止。

這部分我們直接採用 cv2 內建之 `cv2.equalizeHist` 函數實作，執行前後之左右眼影像比較如下所示：

左圖

右圖

均衡前



均衡後



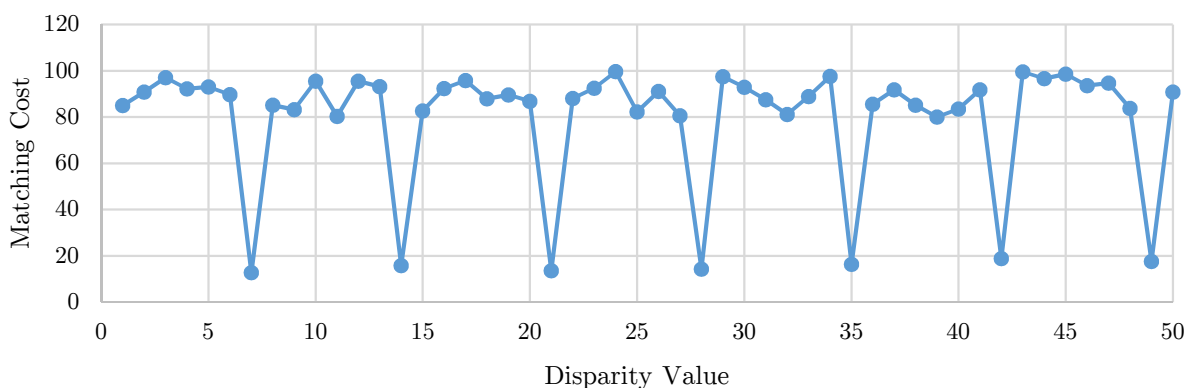
如此可加強配對成本的可信度，且同時並不會影響原始圖像中內容的差異性。

計算配對成本 (Cost Matching)

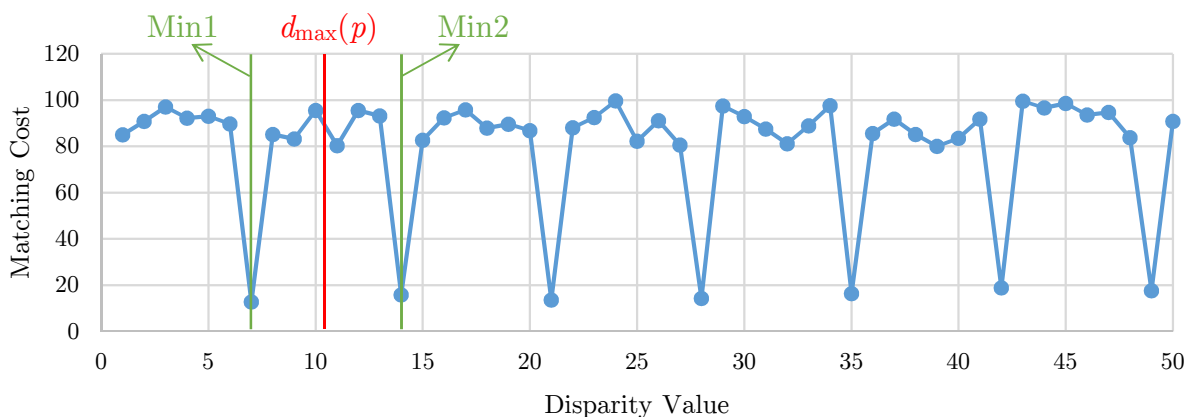
2. 最大視差值決定方法之討論

直覺上，在運程式時，最大視差值之設定應該愈大愈好，如此可以自動涵蓋較大範圍視差之偵測，同時仍可透過每個視差值下算得之配對成本最小值決定最佳結果。然而由於部分測資含有水平方向重複出現的材質或紋理，因此當最大視差值選擇過大時，重複出現之紋理可能每隔一段距離即產生一次配對成本的局部最小值，因此該值的選擇並非愈大愈好。

首先，我們觀察到，當水平方向上有重複條紋出現時，所得之「視差值－配對成本」作圖應和下圖有類似的特徵：



也就是，每隔一段距離（即重複條紋的間距）便會出現一次配對成本之局部最小值。我們採用的方法某種程度上即是在偵測水平方向上重複出現之材質紋理：初始時先不斷加大視差值（直至某一很大的上限 D_{MAX} ），並針對每一個視差值進行配對成本的運算。假設左右眼影像中相距最近的重複條紋（即上圖最靠左邊的最小值）之距離為我們所想取的視差值，則最直覺的方法便是在找到兩或三個相繼出現的局部最小值後，估算其平均間距，並取第一至第二最小值的中點做為該像素 p 應選擇之最大視差值 $d_{max}(p)$ ，如下圖所示。



當每一像素之 $d_{max}(p)$ 皆決定後，整張影像的最大視差值 d_{max}^* 便可擇 $d_{max}(p)$ 之最大值，即如下式：

$$d_{max}^* = \max_p d_{max}(p)$$

然而，上述方法有兩項較強的假設：

- (a) 重複出現的條紋之水平間距須大致相等，亦即 $|\text{Min2} - \text{Min1}| \cong |\text{Min3} - \text{Min2}| \cong \dots$ 。
- (b) 局部最小值必須和其它非最小值有夠大的差距，亦即 $|\text{Avg}(\text{Max}) - \text{Avg}(\text{Min})| \gg 1$ 。

一旦這些條件並未符合，可能導致上述方法所找到的 d_{max} 過小，抑或是迴圈跑到 D_{MAX} 仍然找不到合適的 d_{max} 。更甚者，之所以此方法理論上可行是假設圖片中確實含有重複的條紋；因此，若無重複條紋存在，此方法便無存在的必要性。

經實驗，我們發現本次專題所提供的測資在套用此方法的前後在效果上皆未有明顯的效果差異，然而運算時間上卻多出了數倍（因為尋找 d_{max} 必須花費 $O(D_{\text{MAX}}^2)$ 的時間複雜度）。

在各項因素的考慮下，我們決定不在最終提交的版本採用此方法，然而我們相信上述的討論在合理的條件下使用理應得到較佳的結果。

累積配對成本 (Cost Aggregation)

3. 滑動視窗

一開始實作本次專題時，我們採用的是[1]所提供的十字骨架形可調適大小的視窗。然而，經過多次實驗及參數調整後，發現其產生之結果表現並未較一般之固定大小滑動視窗為佳。因此，最後我們仍然決定在本次專題使用最基本的方形滑動視窗法，惟視窗內每個像素的權重值或有不同。

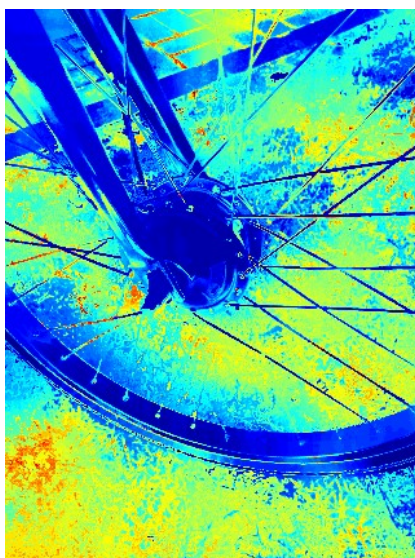
在經驗法則下，我們選擇二維高斯函數視窗來實作，其大小為 9×9 。

視差精化 (Disparity Refinement)

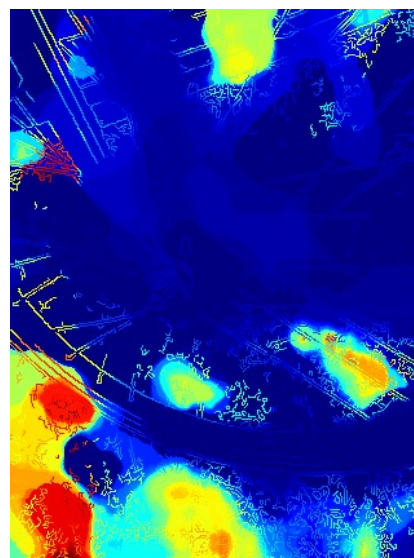
4. 邊緣偵測

此次專題之挑戰之一乃是處理圖片中較為細長的邊界（如腳踏車輪圈之鋼絲以及鐵製購物籃的邊框），而使用視窗配對法較難特別針對這些區域改善。為此，我們將輸入之兩張圖片進行邊緣偵測，並得到只含邊緣資訊的輔助圖；接著，再將兩張輔助圖當作輸入進行視差值估算，結果如下圖所示。

使用原圖之結果



使用僅含邊緣資訊的輔助圖之結果



可從圖中發現，兩張圖分別在不同區域之估算結果各有優劣，其中僅使用邊緣資訊所得的視差值在含有邊界的區域效果較佳。因此，我們決定以分配權重的方式決定哪些區域應該採用哪一張結果之視差值，具體實作方式如下：

$$d(p) = \lambda \cdot d_{\text{edge}}(p) + (1 - \lambda) \cdot d_{\text{normal}}(p)$$

其中， λ 之決定方式為：

$$\lambda = \begin{cases} 0.25, & p \text{ is an edge pixel} \\ 0, & \text{otherwise} \end{cases}$$

5. 視差值區域投票

由於在累積配對成本時使用[1]之方法後，實驗之結果較為不理想，然而其選擇視差值之投票方式確實可以解決大範圍材質相同之背景區域的誤差問題，因此決定在優化的部分使用該方法進行局部區域性的視差值眾數投票。

用於決定初始視差值之普通視窗函數為固定大小，雖可能對各像素給予不同權重，其涵蓋的區域對各像素而言是不變的。另一方面，[1]提供之方法乃是以各像素為中心，向四周延伸尋找彩值相近的像素之集合作為該像素之投票區域。效果上，此方法在大範圍相同背景部分之表現較原始之視窗投票法為佳。

此部分之實作方式與論文中相同，數學上可表示為：

$$d_p^* = \arg \max_d \varphi_p(d), d \in [0, d_{\max}]$$

其中， $\varphi_p(d)$ 代表的是透過[1]所找出之投票區域內、原始視差估計值為 d 的像素票數，而 d_p^* 即是最終最佳化後的視差值。

6. 中值濾波器

如同第四次作業，我們在執行完所有上述的視差精化後，最後仍會經過某固定大小的中值濾波器，去除視差值在部分區域出現的高頻雜訊。雖然概念上很單純，但多次實驗皆表明中值濾波器對最後結果的正確率有相當大的影響。

實驗結果

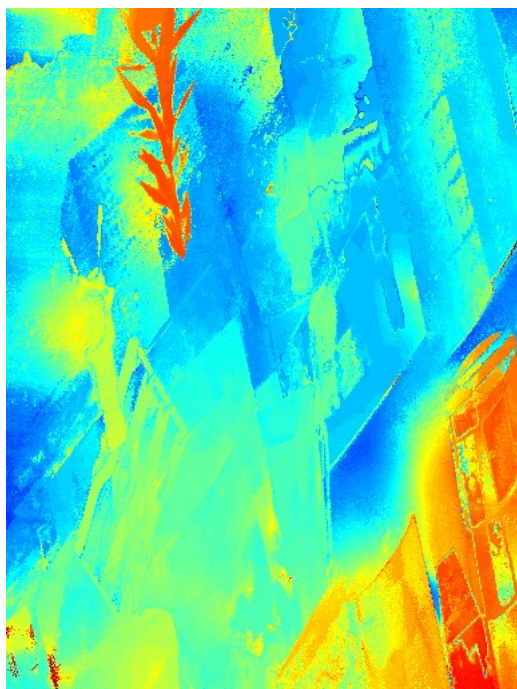
1. 每張圖執行時間約 6 到 10 秒
2. 視差圖之錯誤比率 (Error Rate)

(a) Synthetic

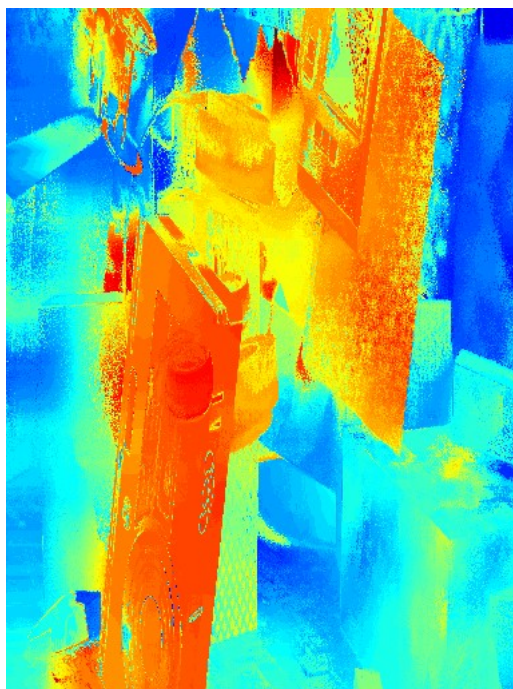
Data	Error Rate	Data	Error Rate
0	3.795786	5	6.144890
1	4.099919	6	3.331357
2	3.848346	7	4.842525
3	5.284838	8	2.703502
4	7.549291	9	5.377114

平均錯誤比率約為 4.697757。

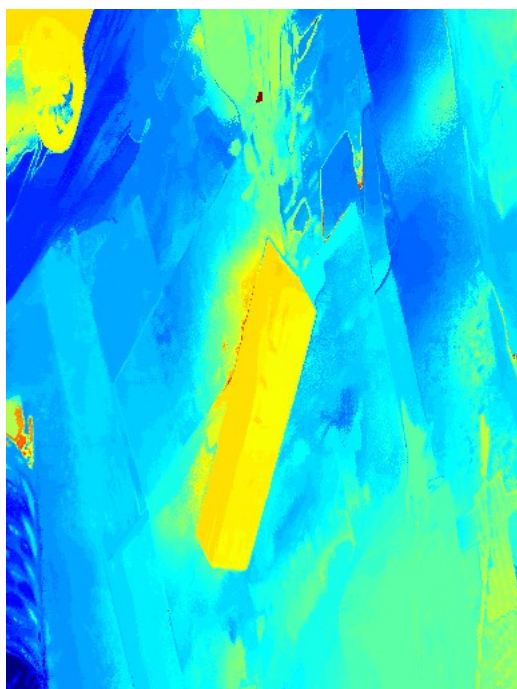
TL0



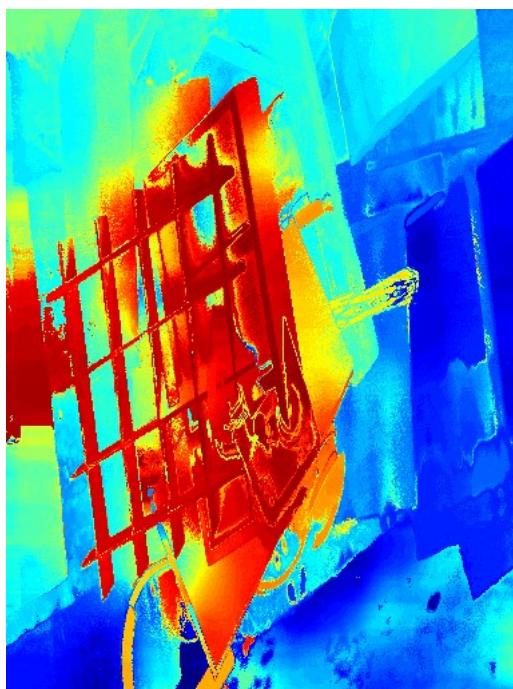
TL1



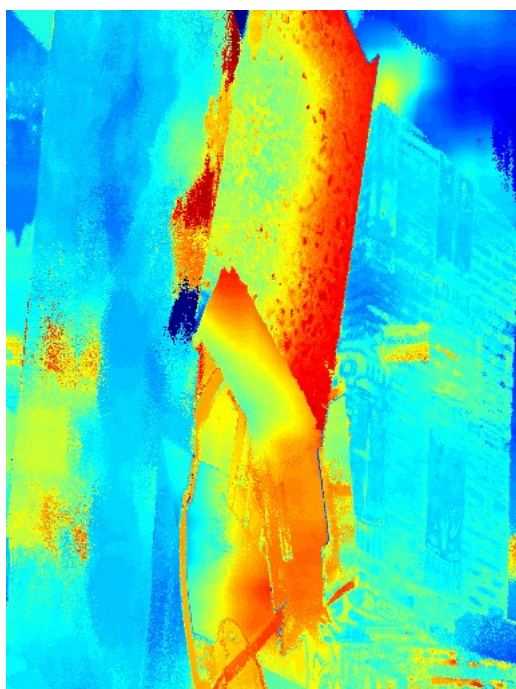
TL2



TL3



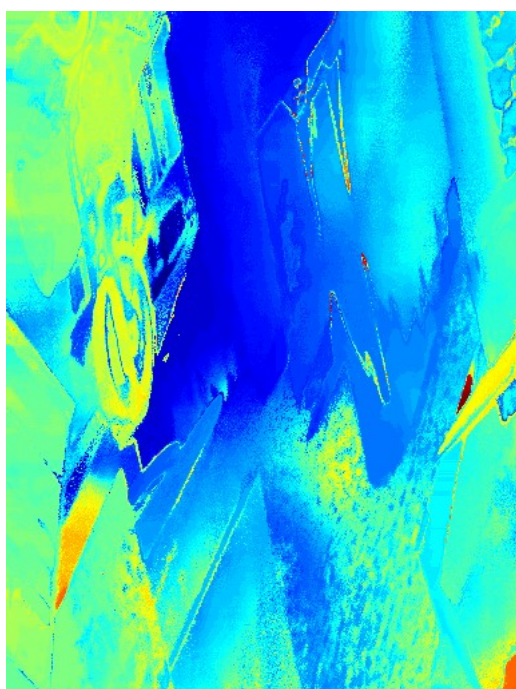
TL4



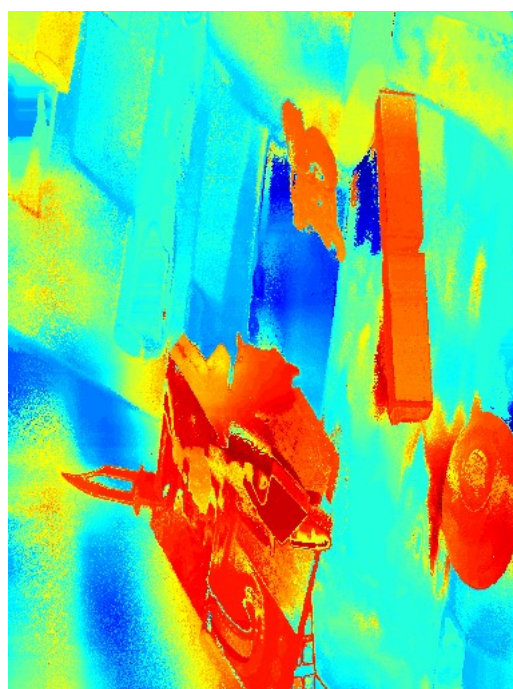
TL5



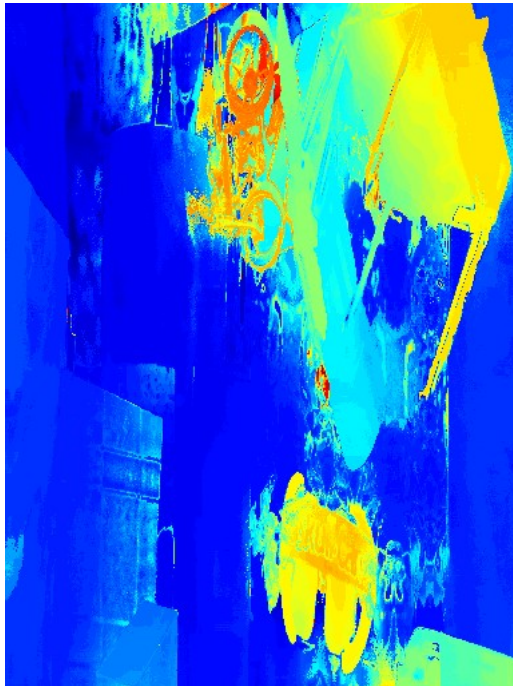
TL6



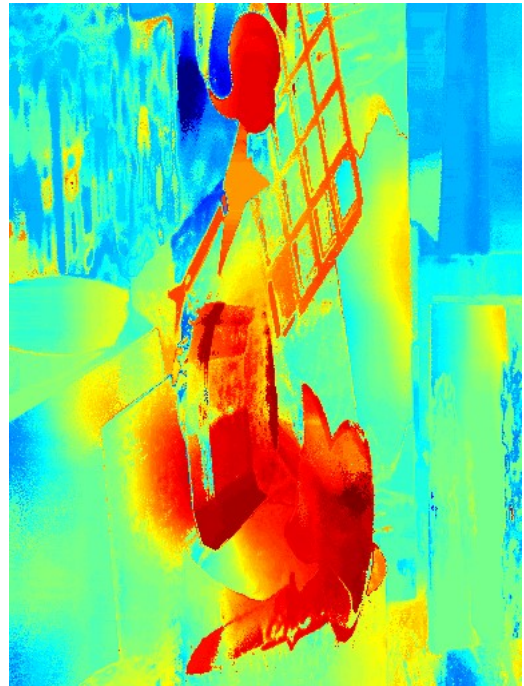
TL7



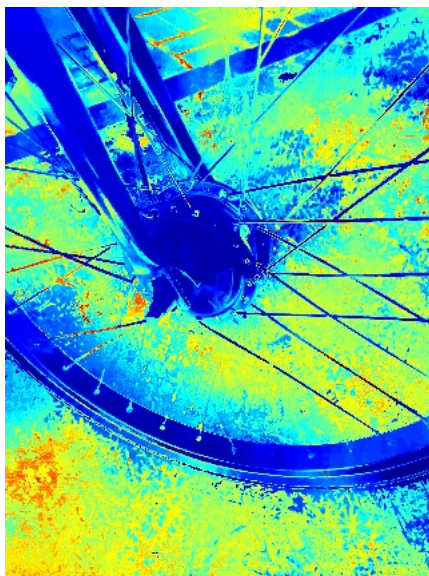
TL8



TL9



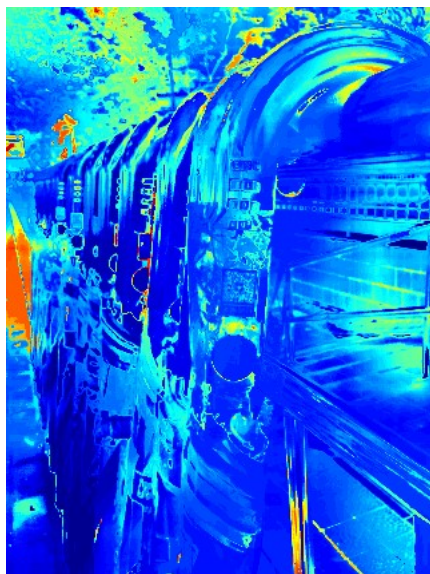
(b) Real
TL0



TL1



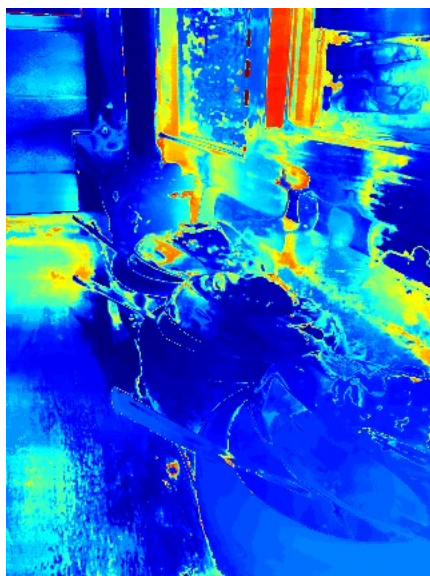
TL2



TL3



TL4



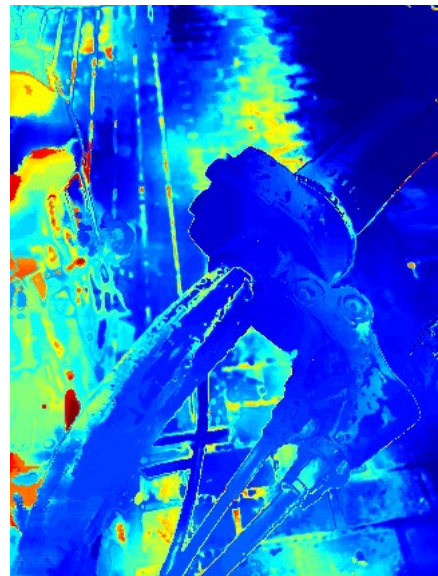
TL5



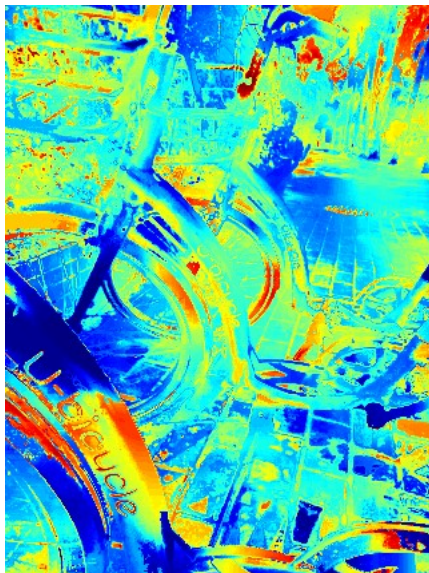
TL6



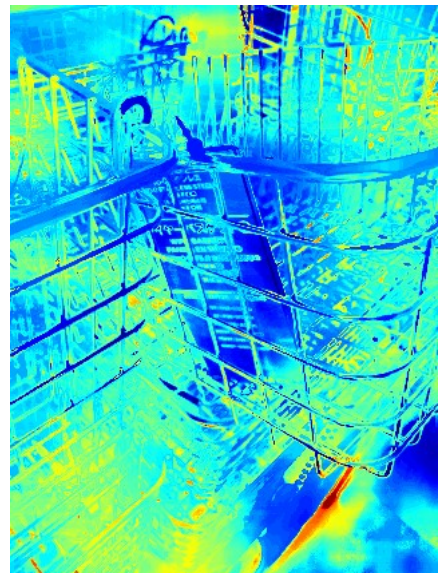
TL7



TL8



TL9



參考資料

1. K. Zhang, J. Lu and G. Lafruit, "Cross-Based Local Stereo Matching Using Orthogonal Integral Images," in *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 19, no. 7, pp. 1073-1079, July 2009.
2. Rhemann et al. Fast cost-volume filtering for visual correspondence and beyond. In CVPR 2011.