

使用影像修補技術消除數位單眼相機感光元件染塵

Using Image Inpainting to Remove Dust Spots from DSLR Images

吳俊霖

中興大學資訊科學所助理教授
e-mail: jlwu@cs.nchu.edu.tw

盧敏曜

中興大學資訊科學所碩士班研究生
e-mail: s9456035@cs.nchu.edu.tw

摘要

隨著單眼相機在消費市場上越來越普遍，使用人數逐年增加，這使得數位單眼相機容易出現感光元件染塵（即 CCD 染塵現象）的問題變得不容忽視。雖然有 CCD 染塵現象的數位單眼相機可以送往廠商以人工手動清除，不過通常價格昂貴，而且一不小心就有可能傷及 CCD，到時候可能更要花一大筆費用來修理，得不償失。最近有一些相機廠商發明出「感光元件自我清理架構」，利用壓電元件產生超音波震動，讓粉塵脫落，不過對於黏度較高的粉塵，則效果不彰。

於是有人提出用傳統去雜訊的方法來將影像上的粉塵清除，由於粉塵雜點比一般雜訊要大得多，因此如果只是使用一般中值濾波器將會失敗。另外，也有人提出利用市面上的影像處理軟體(例如 PhotoImapct)的修容工具來將粉塵拿掉，但是類似這樣的工具，都必須自己手動選取要拿掉的粉塵並自己找材質來填補，因此耗時耗力。而本篇論文的目的則是提出一個能夠自動偵測粉塵的演算法，同時利用影像修補的技術，將影像上令人困擾的粉塵去除，並且維持影像中的材質與線性結構的完整。實驗結果顯示，所提演算法能有效解決數位影像的感光元件染塵問題。

關鍵詞：感光元件染塵、影像修補、材質合成、雜訊去除

that the dust can damage the photos seriously. Cleaning dust on sensors is almost universally warned against by camera makers. If we remove the dust directly by the sensor brush, we might scratch or otherwise damage the cover glass over the sensor, therefore are responsible for the cost of repairs. Camera manufacturers have introduced the dust-reduction solution that involves anti-static coatings and vibration-cleaning of the low-pass filter. However, they can not remove the dust particles with high viscosity effectively.

Some photographers propose to correct this problem in software. The traditional noise reduction methods such as median filter do not perform well in removing the dust spots, since the size of the noise in digital photos caused by dust is large. Some image editing software such as Photoshop also provides the healing brush and clone stamp to stamp out the dust specs, one at a time. However, it is a time-consuming and tedious process if we are taking many photos that suffer from this problem. In this paper, we develop an automatic spots removal algorithm based on the inpainting technique. We first propose a noise detection algorithm to identify the dust spots, it uses Sobel filter and the process is fully automatic. An exemplar-based image inpainting approach is then used to fill holes of dust spots in images, it achieves accurate propagation of linear structures. Several examples on real images are given to demonstrate the effectiveness of the proposed method.

Abstract

The digital cameras are becoming more and more popular nowadays. Digital SLRs not only allow photographers to use inter-changeable lenses to get different ranges of zoom and depth of field, but also give the users more control, it helps to take a better picture. Recently the number of digital SLR users is rising steadily as the equipments drop in price. However, unlike film cameras, current digital SLRs suffer from a frustrating weakness: sensor dust. It is obvious

Keywords: Sensor Dust, Image Inpainting, Texture Synthesis, Noise Removal

1. 前言

數位相機的技術在最近這幾年來有了突破性的進展，由於其省去了傳統相機需要沖洗底片的麻煩，所以使用數位相機的人口正快速的增長中。而隨著數位單眼相機價格下滑，以

及喜愛攝影的人口增加，單眼相機強大且精準的測光性、優異且精準的對焦功能、高感光度以及高擴充性，例如其可更換多種鏡頭，包括廣角鏡頭或長鏡頭，以及可以外接各種閃光燈等等。另外也可以因應自己需求，調整快門時間及光圈大小，種種專業的功能，使得使用單眼相機的人數大為增加。

不過，雖然單眼相機有上述許多優點，但由於數位單眼相機是透過電力操控機械設備，容易產生輕微的靜電，因此在更換鏡頭的時候，若不注意就很容易將小塵埃吸入 CCD 上，所以在拍攝的時候，可能在照片上產生小汙點，這就是所謂的「染塵現象」，如圖 1 所示，(a)為一張有染塵的原始影像；(b)則為(a)的天空局部放大圖，可以發現在天空中有許多顯而易見的雜點。

一般來說，如果單眼相機出現染塵現象，大部分都得送往廠商手動清除，不過通常價格昂貴，而且一不小心就有可能傷及 CCD，到時候可能更要花一大筆費用來修理。最近也開始

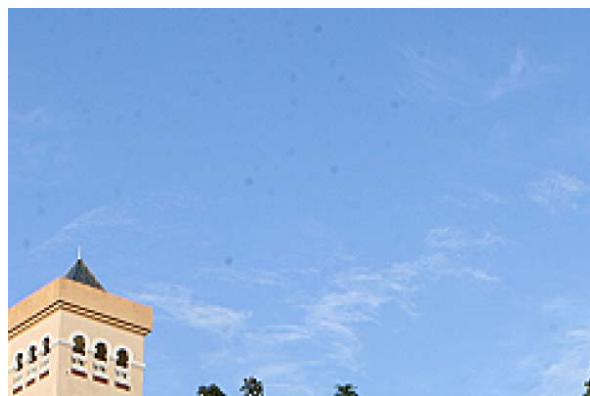
有相機廠商發明出「感光元件自我清理架構」，其原理是使用壓電元件產生超音波震動，讓粉塵脫落，但這樣的機制，可能還是無法對付黏度較高的粉塵，因此在本論文中，我們嘗試利用影像處理技術來解決此問題。

首先，如果利用一般去雜訊的方法，例如：中值濾波器(median filter)，雖然對於一般的脈衝雜訊能夠處理的很好，但是由於大部份的粉塵所造成的雜點都較大，所以利用傳統去雜訊的方法將會失敗，如圖 2 所示，(a)為一張有脈衝雜訊的影像；(b)是利用中值濾波器去脈衝雜訊後的結果，可以發現效果很不錯。而圖 3 所示為圖 1(a)有 CCD 粉塵雜點的影像經過大小為 3×3 中值濾波器去除雜訊的結果。我們可以發現，天空中仍然有許多粉塵殘留，而且影像的細節也被模糊化了。

另外，也有人提出利用 PhotoImpact 的修容工具來將粉塵拿掉，其用法是先在未染塵的影像區域中選定要填補粉塵雜點的材質，接著再利用滑鼠在粉塵雜點的位置上拖曳，這樣子



(a)



(b)

圖 1 (a)有 CCD 染塵現象的影像；(b)為(a)的天空局部放大圖



(a)

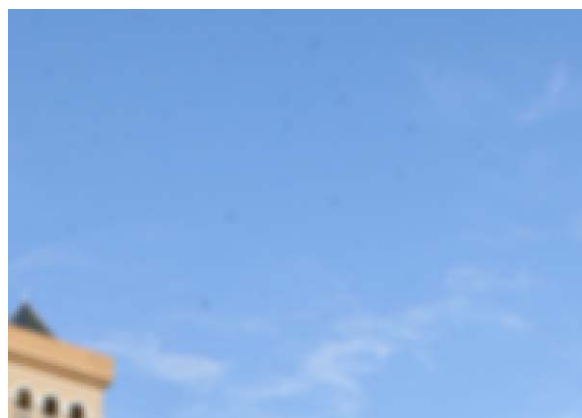


(b)

圖 2 (a)一張有脈衝雜訊的影像；(b)以大小為 3×3 中值濾波器去雜訊後的結果



(a)



(b)

圖 3 (a)以大小為 3×3 中值濾波器對圖 1(a)影像去除粉塵的結果；(b)為(a)的天空局部放大圖

就能把剛剛所選定區域上的材質填補進去而將粉塵雜點移除，但是缺點為必須自己手動選取要拿掉的粉塵並自己找材質來填補，所以耗時耗力。除此之外，PhotoImpact 的修容工具主要是利用材質複製方法，將周圍的影像資訊貼補到有染塵的地方。因此，如果粉塵是落在影像中線性結構的地方，則利用修容工具修補的效果將會不是很好。所以在本論文中，我們提出以 patch 為主的影像修補(image inpainting)策略來解決此問題。我們先從一張影像上自動的偵測單眼相機內 CCD 被染塵的位置並標示起來，然後進行修補，其優點是能夠全自動化的針對有染塵的影像進行修補且即使粉塵落在影像中線性結構的地方，依然可以得到很好的結果。

這篇論文我們將以下列結構循序探討：第二節我們將簡短回顧一下有關於影像修補的技術種類及其方法，第三節則是說明所提自動偵測出粉塵的演算法，第四節我們會說明用以

填補粉塵的演算法細節，第五節則是實驗結果與討論。實驗結果顯示不管粉塵落在任何位置我們都可以將其移除，並不會破壞影像的結構。第六節則是結論與未來展望。

2. 影像修補

影像修補(image inpainting)是專門用來修補被破壞影像的一項技術，其應用的範圍包括舊照片上裂痕的修補、移除不想要的物件以及附加在圖片上的文字、戳印，尤其是對於極具保存價值的書法字帖及畫作做數位典藏時，我們都可以用影像修補的技術來處理可能出現的刮痕、裂縫或是污損，以期達到完善的保存。

在之前的文獻中，影像修補的技術主要可以大略分為兩種類型。一種是利用材質合成(texture synthesis) [1],[2],[3] 的方法，材質合成的方法可以用來修補大範圍的目標區域，但是卻只適合修補影像中純材質的部份，如圖 4 所示，我們只需要利用草地的材質來填補影像下方鴿子的部份，所以修補的結果很好。然而該方法對影像中具有線性結構的部份，卻難以維持其連續性，主要因為其在材質貼補的順序，大部分材質貼補的順序都是從左而右、從上而下(raster-scan order)或是以同心圓的順序，也因此造成線性結構的遺失。

另外一種方法則是在 2000 年由 Bertalmio 等人率先提出基於偏微分方程(Partial Differential Equations)的影像修補演算法[4]。其首先在已經標出要填補的區域，定義出其區域的輪廓，並沿著輪廓計算要填補區域外的梯度向量(gradient vector)，然後利用 Anisotropic Diffusion 的方法沿著垂直其梯度向量的方向

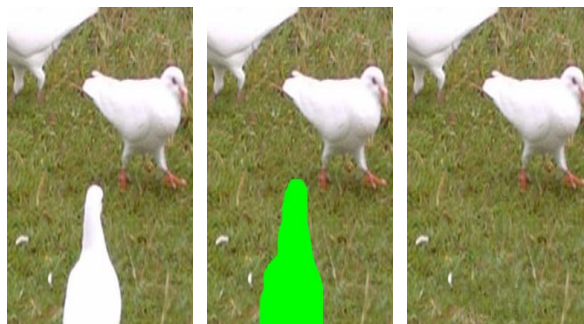


圖 4 用材質合成的影像修補結果



圖 5 用偏微分方程的影像修補的結果

(即等照度線(isophote)的方向)將要填補區域外的資訊擴展進去,其目的就是要能夠維持線性結構的連續性,如圖 5 所示,影像中有被畫筆亂畫的痕跡,利用偏微分方程修補後,可以發現效果很好且影像中線性結構的部份也被維持得很完整。但也由於擴散(diffusion)的關係,所以其在填補大區域範圍的時候,會有很明顯的模糊影像產生,因此這一類的方法比較適合填補影像上的小裂縫、刮痕、文字或是筆觸之類。

之後基於 Bertalmio 所提出的概念,也發展出許多類似的方法,包括 2001 年由 Oliveira 等人提出的快速數位影像修補(Fast Digital Image Inpainting) [5],其方法大大加快了影像修補的速度,另外還有在 2001 年由 Chan 與 Shen 所提出的 Curvature-Driven Diffusions (CCD)作非材質的影像修補演算法 [6],以及 2004 年由 Alexandru Telea 所提出的 Fast Marching Method (FMM) [7],其方法的優點是簡單實作,並且又有效率,而且也能得到很好的結果。不過這些方法也都只僅限於修補影像上的小裂縫、刮痕、文字或是筆觸等,對於影像上大範圍的修補區域仍然會產生模糊。

因為大部分真實世界的影像,不會均是由純材質或是純結構所構成的,因此在之後就有許多人提出結合材質合成及偏微分方程的兩者方法的優點來修補影像。2003 年由 Bertalmio 等人所提出的方法[8],是將原來的影像先分解成材質及結構兩個部份,材質的部份就利用材質合成的方法去修補;而結構部份則利用偏微分方程的方法去修補,最後的結果則是將兩個部份合起來得出。然而這個方法依然僅限於修補小範圍的區域且依然由於擴散的關係,仍會模糊要填補的區域。同樣在 2003 年,Yamauchi 等人[9]提出將影像利用離散餘弦轉換(DCT)計算出影像的低頻及高頻兩個部份,然後利用

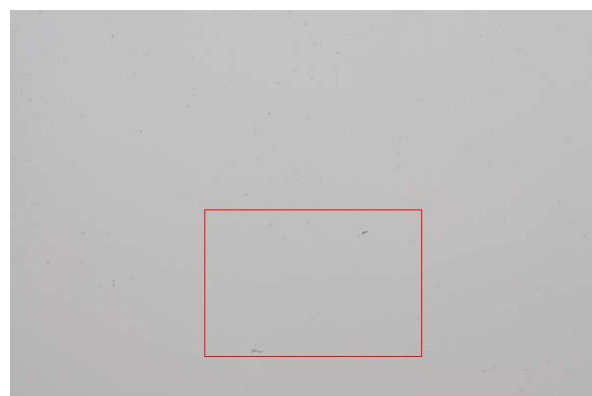
偏微分方程的方法修補低頻部份,利用多重解析度(multiresolution)的材質合成方法去修補高頻部份,最後的結果同樣把兩個部份合起來得出。另外,一樣在 2003 年由 Jia 等人所提出的演算法[10],是將不同的材質部份作分割(segmentation),然後利用 tensor-voting 來維持線性結構的連結,這個方法對於曲線結構的維持有很好的結果,但是卻需要額外的影像分割的步驟,以及必須決定在兩種材質邊界之間該用什麼來組成。

在 2004 年,由 Criminisi 等人提出的「基於範例影像修補(exemplar-based image inpainting)」演算法[11],其方法是先找出要填補區域的輪廓,然後沿著輪廓的每個像素,以其為中心定義一個 9×9 大小的 patch,然後計算每個 patch 的優先權(priority)大小,有最大優先權的 patch 則優先填補。而優先權的計算則是利用 confidence term 與 data term 相乘。其中,confidence term 是用來計算可信賴的像素有多少,而 data term 則是用來計算垂直輪廓邊緣的單位向量與等照度線(isophote)的內積值,目的是為了在做材質合成時,給予落在線性結構的 patch 較大的優先權,使其優先填補,以確保線性結構的完整。之後,Sun 等人提出以結構擴展(structure propagation)來作影像修補的演算法[12],其方法是先定義一條或多條曲線從影像已知區域延伸到要被填補的區域以模擬影像中所遺失的結構,接下來在要被填補區域的曲線上稀疏取樣以產生固定點(anchor points),然後以固定點為中心選取一個 patch,再從已知影像區域的曲線上選取 patch 比對以找出最相似的 patch 作填補。在結構填補完後,剩餘的區域再利用材質合成的方式將其補滿。此方法雖然能夠將影像中的材質與結構維持得很完整,但是必須由使用者手動定義出曲線來模擬影像中遺失的結構,所以其演算法並非全自動的。

而我們在本論文中針對修補 CCD 上粉塵所提出以 patch 為主的影像修補策略,即是採用由 Criminisi 等人提出的「基於範例影像修補」演算法[11]。其原因是這個方法能夠用單一的機制就能同時把影像中的材質及線性結構部份修補完整,而且非常有效率,也不需要使用者複雜的去描繪曲線結構,所以我們希望利用這個方法來去除 CCD 染塵的問題。

3. 自動粉塵偵測

首先，為了方便偵測粉塵，我們將有染塵的單眼相機，使其光圈開到最小，並對著一張白紙或任何全白的物體拍攝，如圖 6 所示，(a) 為對著白紙拍攝的影像；(b) 則為(a)的局部放大圖，可以發現影像中有許多清楚可見的粉塵雜點。



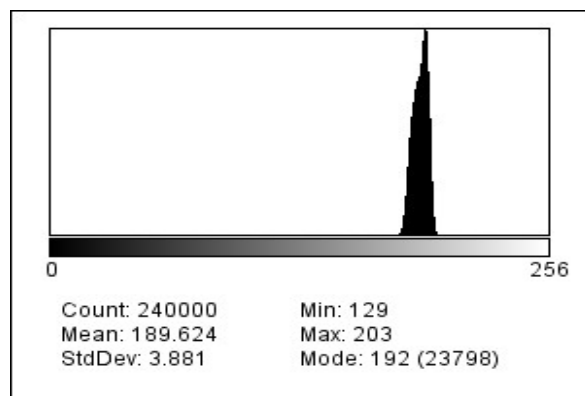
(a)



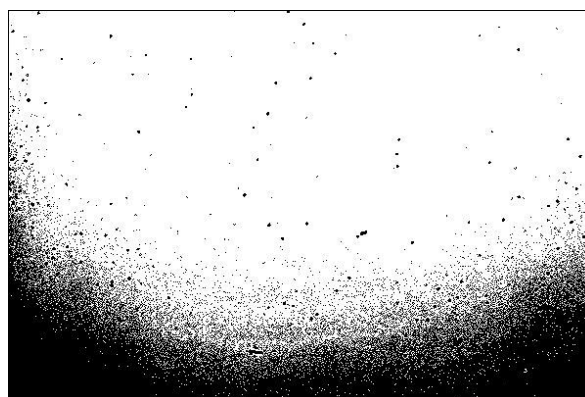
(b)

圖 6 (a)對著白紙拍攝的影像；(b)為(a)的局部放大圖

接著要偵測粉塵，也就是要將粉塵雜點從背景分離出來，而最簡單的方法就是直接對影像取臨界值(threshold)。於是，我們觀察圖 6(a) 的直方圖(histogram)分佈，如圖 7(a)所示。為了盡可能把粉塵偵測出來，因此我們取臨界值 190 將圖 6(a)作影像二值化，如圖 7(b)所示，我們可以發現，整個影像下方將隨著臨界值取的越大，干擾粉塵偵測的區域範圍也越大。主要是因為一般影像都會有亮度(intensity)上的變化，因此直接對影像取臨界值做二值化以偵測粉塵的效果並不好。



(a)



(b)

圖 7 (a)圖 6(a)的直方圖分佈；(b)直接對圖 6(a) 設臨界值(threshold = 190)將影像二值化的結果

所以，在本論文中我們提出利用 Sobel 運算子來克服影像裡有亮度變化的方法以自動偵測粉塵。由於粉塵雜點在影像中其亮度相對是較小的，因此我們可以利用一階微分，也就是梯度運算(gradient operation)，將粉塵雜點的邊緣偵測出來。原理就是利用梯度運算能夠增強微小且不連續點的能力以及能夠除去緩慢變化的背景。而 Prewitt 與 Sobel 則是實際計算梯度時最常使用的運算子，其中 Sobel 運算子藉由給予中心點較大的加權值來達到平滑的特性，如圖 8 所示，因此其相對於 Prewitt 運算子有較優越的抗雜訊能力，所以本論文我們是利用 Sobel 運算子來偵測影像中的粉塵邊緣。

-1	-2	-1
0	0	0
1	2	1

-1	0	1
-2	0	2
-1	0	1

圖 8 Sobel 運算子

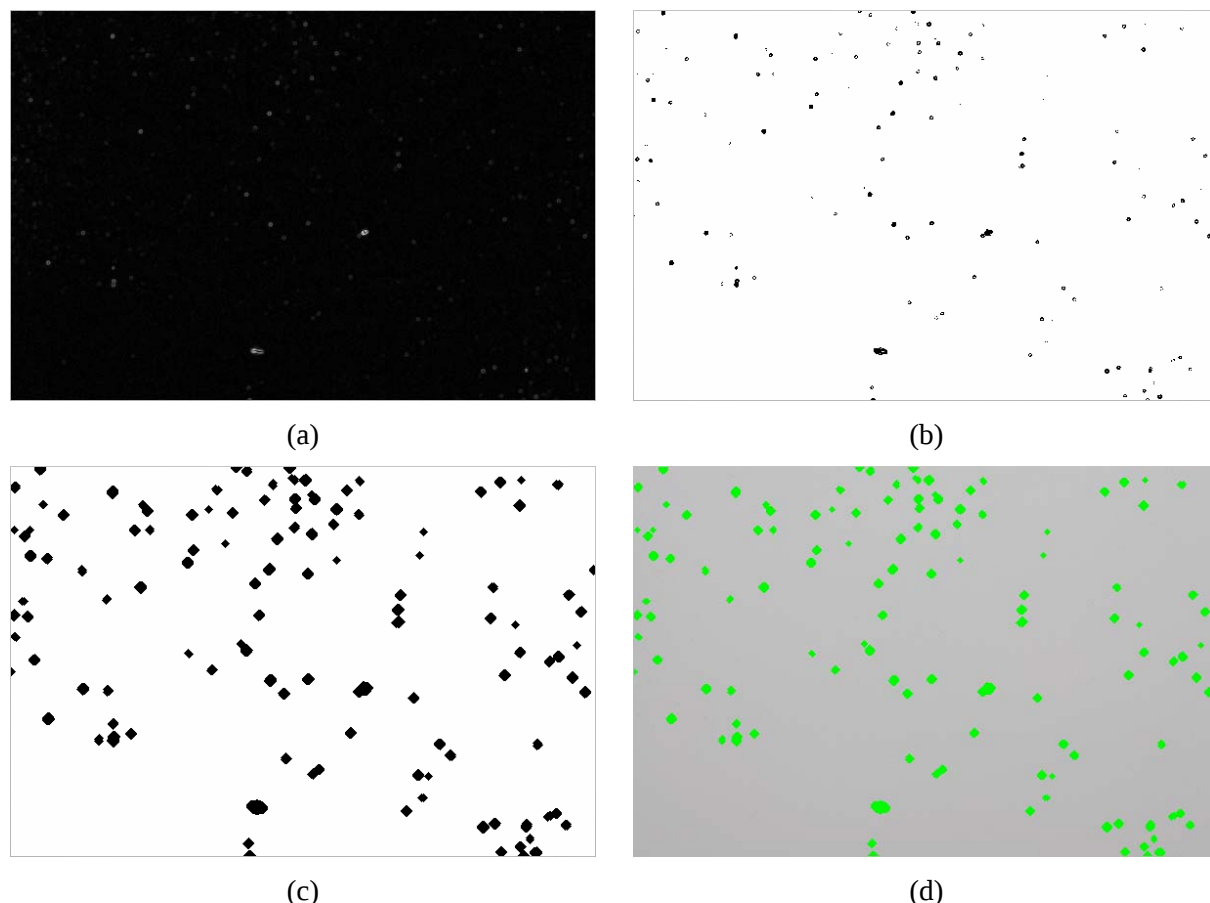


圖 9 (a)將 Sobel 運算子套用在圖 6(a)所得到的結果；(b)對(a)取臨界值(threshold = 27)做影像二值化的結果；(c)對(b)做擴張運算後的結果；(d)利用(c)所得遮罩將圖 6(a)的粉塵雜點標記起來

在粉塵邊緣偵測出來之後，基於我們是要移除整個粉塵，因此我們利用形態學影像處理(morphological image processing)上的擴張運算(dilation operation)來將粉塵補滿以利之後移除的動作。如圖 9 所示，(a)是套用 Sobel 運算子在圖 6(a)將粉塵雜點邊緣偵測出來的結果；(b)則是在(a)取一個臨界值做影像二值化的結果，而此張影像的臨界值我們取 27；(c)則是針對(b)的結果作擴張運算，將原本結構不完整的粉塵雜點都填補起來，以利之後將粉塵從影像中移除。由於在影像上會出現粉塵雜點是因為在相機裡頭的 CCD 染塵，所以出現在影像上的粉塵位置都是固定的，因此圖 9(c)就可以作為我們的遮罩(mask)，針對所有同一台相機所拍攝的影像，都可以套用同一張遮罩將影像上頭的粉塵雜點標記起來，如圖 9(d)所示，其結果即為利用圖 9(c)所得到的遮罩將圖 6(a)的粉塵雜點標記起來。

4. 基於範例影像修補演算法

接下來，我們將利用 Criminisi 等人[11]提出的「基於範例影像修補」演算法來移除 CCD 或 CMOS 染塵現象，其示意圖如圖 10 所示。首先，我們先將影像中被染塵部份標記出來，而它就是我們將要修補的目標區域(Ω)，並把目標區域 Ω 的輪廓定義出來，我們把它標記為 $\partial\Omega$ ，之後在重複以下步驟：

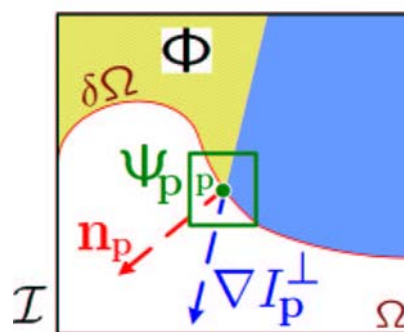


圖 10 基於範例影像修補的符號示意圖[11]

- (1) 初始化。在初始化的時候，會先給予每個像素一個 confidence value，以 $C(p)$ 表示，其規則是假如 p 點是在目標區域裡，則 $C(p)=0$ ，否則 $C(p)=1$ 。
- (2) 對於每個位在 $\partial\Omega$ 上的點 p ，我們定義一個以 p 點為中心的 9×9 大小的 patch，以符號 ψ_p 表示。之後計算每個位在輪廓上的 patch 優先權，計算公式為： $P(p)=C(p) \times D(p)$ ，其中 $C(p)$ 為 confidence term，計算方式如下：

$$C(p) = \frac{\sum_{q \in \Psi_p \cap (I - \Omega)} C(q)}{|\Psi_p|} \quad (1)$$

其中， $|\psi_p|$ 則是 patch ψ_p 的面積大小。而 $D(p)$ 為 data term，計算方式如下：

$$D(p) = \frac{|\nabla I_p^\perp \cdot n_p|}{\alpha} \quad (2)$$

其中， $\nabla I_p^\perp$ 為等照度線，即垂直於梯度向量的向量， n_p 為垂直於輪廓的單位向量，而 α 為 255，對 $D(p)$ 做正規化的動作。

- (3) 選擇一個有最高優先權的 patch (以符號 $\psi_{\hat{p}}$ 表示) 來被填補，然後在來源(source)區域(Φ)，針對 9×9 大小的 patch (以符號 ψ_q 表示)，利用 RGB 三色的 Sum of Squared Differences (SSD) 來比對兩個 patch 相似程度，公式如下：

$$SSD = \sum \left\{ \begin{aligned} &(R_{\hat{p}}(p) - R_q(p))^2 + \\ &(G_{\hat{p}}(p) - G_q(p))^2 + \\ &(B_{\hat{p}}(p) - B_q(p))^2 \end{aligned} \right\} \quad (3)$$

其中， $R_{\hat{p}}$ 、 $G_{\hat{p}}$ 、 $B_{\hat{p}}$ 分別表示 patch $\psi_{\hat{p}}$ 內像素 p 的 RGB 三色，而 R_q 、 G_q 、 B_q 則代表表示 patch ψ_q 內像素 p 的 RGB 三色。而 SSD 值最小的那塊 patch 即與 patch

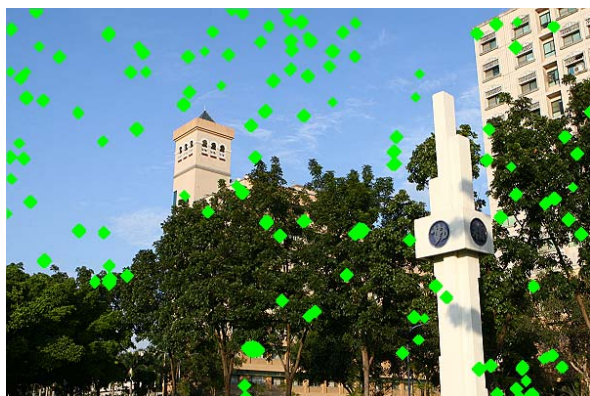
$\psi_{\hat{p}}$ 最為相似的 patch (以符號 $\psi_{\hat{q}}$ 表示)。找出最相似的 patch 之後，就針對 patch $\psi_{\hat{p}}$ 未填滿的像素對應於 patch $\psi_{\hat{q}}$ 的像素位置，一個像素對一個像素貼補過去即可。

- (4) 更新剛剛被貼補區域的 confidence value。更新方式為把步驟(2)計算出來的 $C(p)$ 值給予剛被貼補的每個 pixel，作為它們的 confidence value。
- (5) 如果目標區域 Ω 存在，則重複步驟(2)~(4)，直到整各區域都被填滿為止。

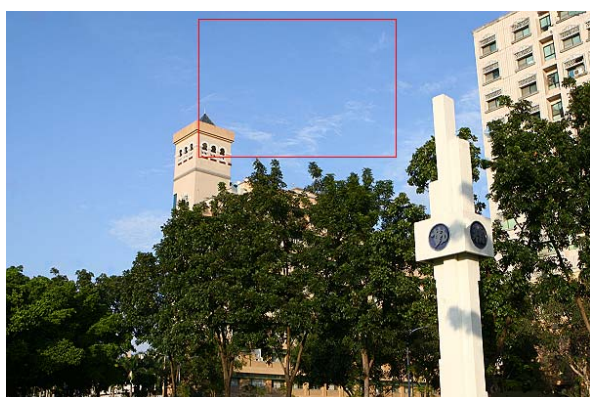
我們之所以會利用 Criminisi 等人提出的「基於範例影像修補」演算法[11] 來修補粉塵，主要是因為假如粉塵落在影像中有線性結構的地方，那麼這個方法一樣可以修補的很好，這都是因為此演算法在計算 patch 的優先權時，加入了 data term 這一項，使得影像中的線性結構會優先被填補，以維持線性結構的完整性。

5. 實驗結果與討論

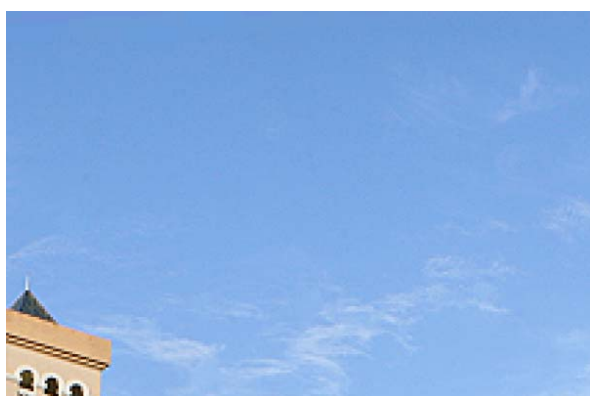
首先，我們觀察圖 1(a)被修補過後的結果，如圖 11 所示，(a)為利用圖 8(c)所得到的遮罩將圖 1(a)的粉塵雜點標記起來，我們可以發現粉塵雜點不只落在天空中，也落在右邊大樓的窗戶結構上以及影像中許多線性結構的地方；(b)為利用「基於範例影像修補」演算法[11]來作修補的結果，我們發現不但天空變得非常潔淨，而且影像中所有物體的結構都保持的很好；(c)則為(b)修補過後的天空局部放大圖，我們可以跟圖 1(b)做比較，很明顯地，該演算法可以將天空中的粉塵雜點有效地移除。



(a)



(b)



(c)

圖 11 (a)套用圖 9(c)所得遮罩把圖 1(a)的粉塵標記起來；(b)利用「基於範例影像修補」演算法[11]來作修補的結果；(c)為(b)的天空局部放大圖

接著，如圖 12 所示，(a)為一張有染塵的原始影像；(b)為(a)的天空局部放大圖，可以發現有很多顯而易見的粉塵雜點；(c)則是(b)修補過後的結果，可以發現所有粉塵雜點均被有效地移除。



(a)



(b)



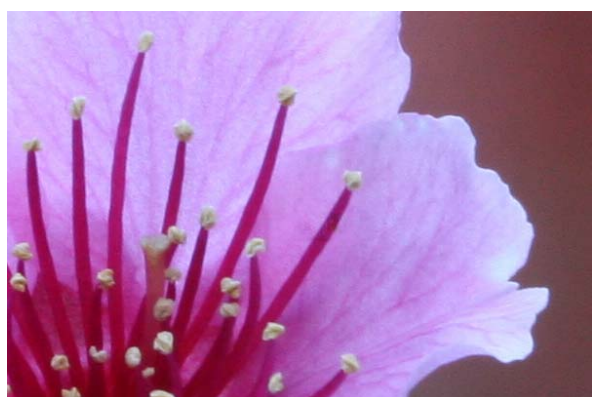
(c)

圖 12 (a)有染塵的原始影像；(b)為(a)的天空局部放大圖；(c)為(b)修補過後的結果

為了測試所提演算法在修補含線性結構的粉塵雜點的可靠性，我們以圖 13 所示的影像來驗證之。圖 13(a)所示為一張粉塵雜點正好落於花蕊憲性結構的影像；(b)則是修補過後的結果。很明顯地，這說明了所提演算法不但能夠將粉塵移除而且也能維持線性結構的完整性。



(a)



(b)

圖 13 (a)粉塵雜點落在花蕊的線性結構上；
(b)為(a)修補過後的結果

由粉塵是落於感光元件，即 CCD 或 CMOS 上，因此由同一台數位單眼相機所攝得的影像之粉塵雜點位置都是固定的。所以我們只需要針對其中一張影像，將其粉塵雜點偵測出來，作為我們的遮罩，之後只要是該相機所拍攝的影像，那麼都可以套用同樣的遮罩，大量批次地將影像上頭的粉塵雜點位置標記起來以進行修補，而且整個過程都是全自動的，對於以往耗時耗力的人工手動移除粉塵雜點，我們所提出來的方法不但有效而且也大大節省了時間與人力。

6. 結論與未來展望

CCD 染塵可以說是使用單眼相機最大的困擾，而且其粉塵在天氣良好時所拍攝的照片尤其明顯。此外，在我們所拍攝的照片當中可以發現，粉塵比一般傳統雜訊要大得多，因此利用一般去雜訊的方法是不可行的，而且當

CCD 被染塵的部份一多，用 PhotoImpact 的修容工具手動清除粉塵，可以說是耗時耗力，而且每張照片都得重複作一次。

所以在本論文中，我們提出一個自動偵測粉塵的演算法，其結合了 Sobel 運算子以及形態學的擴張運算，而經由實驗證明，此演算法可以有效地將粉塵偵測出來，作為之後進行影像修補所使用的遮罩。接著，只要是同一台照相機所拍攝的影像，那麼都可以套用同樣的遮罩，將影像上頭的粉塵雜點位置標記起來，再利用「基於範例影像修補」演算法去修補影像中被標記起來的地方，而經由修補過後的結果證明，不論粉塵落在天空中或是物體的結構上，都能被修補的很好。而從粉塵偵測到進行影像修補的整個過程均是全自動的，因此大大改善了以往在處理感光元件染塵問題的不便性。

參考文獻

- [1] M. Ashikhmin, "Synthesizing natural textures," in *Proc. ACM Symp. Interactive 3D Graphics*, Mar. 2001, pp. 217–226.
- [2] A. Efros and T. Leung, "Texture synthesis by nonparametric sampling," in *Proc. Int. Conf. Computer Vision*, Kerkyra, Greece, Sept. 1999, pp. 1033–1038.
- [3] A. Efros and W. T. Freeman, "Image quilting for texture synthesis and transfer," in *Proc. ACM Conf. Computer Graphics (SIGGRAPH)*, Aug. 2001, pp. 341–346.
- [4] M. Bertalmio, G. Sapiro, V. Caselles, and C. Ballester, "Image inpainting," in *Proc. ACM Conf. Comp. Graphics (SIGGRAPH)*, New Orleans, LA, July 2000, pp. 417–424.
- [5] M. M. Oliveira, B. Bowen, R. McKenna, Y.-S. Chang, "Fast Digital Image Inpainting," in *Processing of the International Conference on Visualization, Imaging and Image Processing (VIIP 2001)*, 2001, pp. 261–266.
- [6] T. F. Chan and J. Shen, "Non-texture inpainting by curvature-driven diffusions (CDD)," *J. Vis. Comm. Image Rep.*, vol. 4, no. 12, pp. 436–449, 2001.
- [7] A. Telea, "An Image Inpainting Technique Based on the Fast Marching Method," *Journal of Graphics Tools*, 2004, pp. 25–36.
- [8] M. Bertalmio, L. Vese, G. Sapiro, and S.

- Osher, "Simultaneous structure and texture image inpainting," *in Proc. Conf. Comp. Vision Pattern Rec.*, Madison, WI, 2003, pp. 882–889.
- [9] H. Yamauchi, J. Haber, H.-P. Seidel, "Image Restoration Using Multiresolution Texture Synthesis and Image Inpainting," *Computer Graphics International*, 2003, pp.108-113.
- [10] J. Jia and C.-K. Tang, "Image repairing: Robust image synthesis by adaptive nd tensor voting," *in Proc. Conf. Computer Vision and Pattern Recognition*, Madison, WI, 2003, pp. 643–650.
- [11] A. Criminisi, P. Pérez and K. Toyama, "Region Filling and Object Removal by Exemplar-Based Image Inpainting," *IEEE Trans. Image Processing*, Vol. 13, No. 9, 2004, pp.1200-1212.
- [12] J. Sun, L. Yuan, J. Jia, and HY Shum, "Image Completion with Structure Propagation" *in Proceedings of SIGGRAPH* 2005, pp. 861-868.