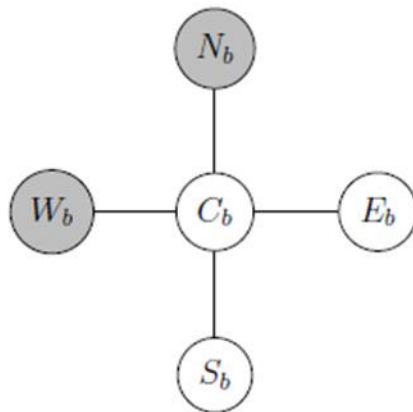
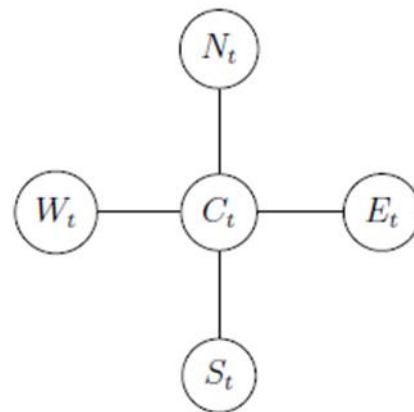


一開始我利用講義上提供的 Poisson editing eq.



(a) The values to be solved.



(b) The corresponding target image.

$$C'_b = \frac{1}{4} \left[\underbrace{4C_t - (N_t + W_t + S_t + E_t)}_{\text{Fixed during iterations}} + \underbrace{(N_b + W_b)}_{\text{Current value}} + (S_b + E_b) \right]$$

發現雖然到最後可以達到收斂，但是 iteration 的次數太多

因此我實現了講義上提供的加速方法 Successive Over-Relaxation method

$$C'_{b,SOR} = \omega C'_b + (1 - \omega)C_b.$$

大幅度降低 iteration 的次數和需要的時間

計時的 timer 我是加在 SOR 開始前和 iteration 結束後

```
//timer start    //print timer
Timer timer;     timer.Pause();
timer.Start();   printf_timer(timer);
```

ω	SOR Iterations	Normal iterations	Execution Time(us)
1.5	10	15000	6199232
2	10	9000	3593112
2.5	10	15000	667003

由跑出來的 **data**，我們可以發現當 ω 越大的時候，執行時間會減少，以 2 為分水嶺，越大反而不會收斂，另外由於我們觀察影像是用肉眼去看，很難定義一個標準叫做完全縫合，所以多少會造成一些誤差。