

Pràctica 6 : Stereo Matching

Question 1. Complete the 'stereo_computation' function with the computation of the SSD cost

```
if strcmpi(cost_function,'SSD') %Sum of Squared Differences
    error = Inf;
    for k = kmin:kmax

        windowr = double(Ir(i-w_step:i+w_step, k-w_step:k+w_step,:));
        costSSD = sum(sum(sum(weight.*(windowl-windowr).^2)));

        if(costSSD<error) %if the current cost is smaller than the previous one
            error = costSSD; %we refresh the error
            kbest = k; %and we choose it as the best candidate
        end
    end
end
```

Question 2. How do the error and the occluded areas are related?

Donades dues imatges (I i I') que capturen dos punts de vista diferents d'un mateix objecte, cap la possibilitat que hi hagi punts P d'aquesta escena que poden ser representats en una imatge (I), pero no en l'altra (I'). Això dóna lloc a oclusions en una de les imatges (I), i per tant provocarà un augment de l'error, ja que no serà possible trobar la correspondència p' real d'aquest punt p en l'altra imatge (I').

Question 3. Compute the global error

L'error global l'hem calculat amb les següents línies de codi:

```
error = abs(double(Igt(18:h-18,18:w-18))-disparity(18:h-18,18:w-18)*255/max_dis);
[h,w] = size(error);
globalError = sum(sum(error))/(h*w);
```

Question 4. Evaluate the results changing the window size (e.g. 3x3, 9x9, 21x21, 31x31). Comment the results and differences in the results.

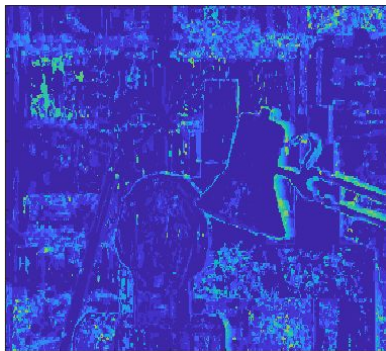
Com més petita és la finestra, més detalls es poden representar, però a la vegada es genera més soroll. Inversament, com més gran és la finestra que fem servir, més suau serà el mapa de disparitat, però es perdrà més detall i precisió.

A mesura que hem anat augmentant la finestra, l'error global s'ha anat fent cada vegada més petit, fins a arribar al 12.2292 amb una finestra de 9x9.

A continuació, presentem una taula amb els resultats obtinguts utilitzant SSD cost amb finestres de diferents mides:

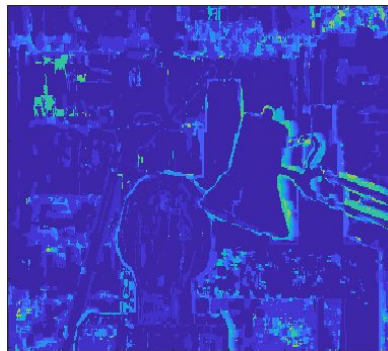
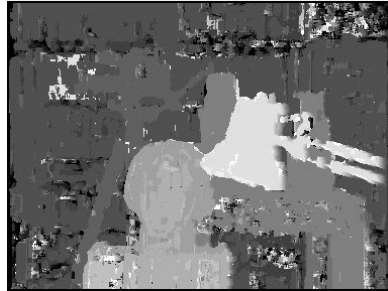
$ws_{\text{ize}} = 3 \times 3$

Global Error = 19.7862



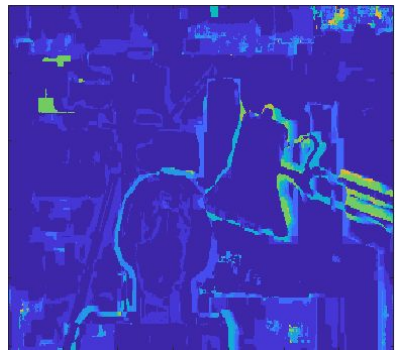
$ws_{\text{ize}} = 5 \times 5$

Global Error = 14.9974



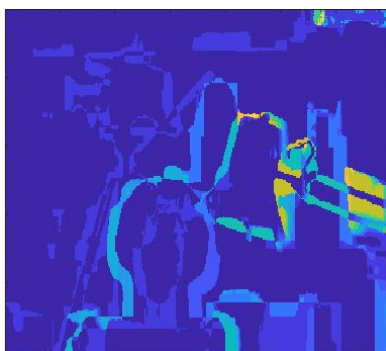
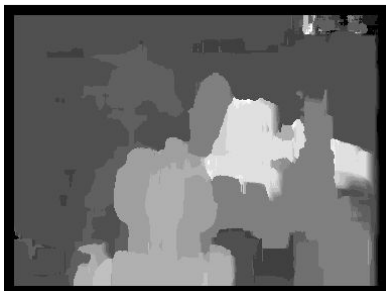
$ws_{\text{ize}} = 9 \times 9$

Global Error = 12.2292



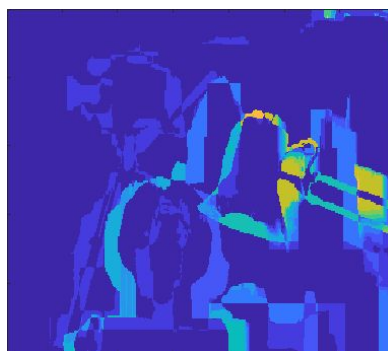
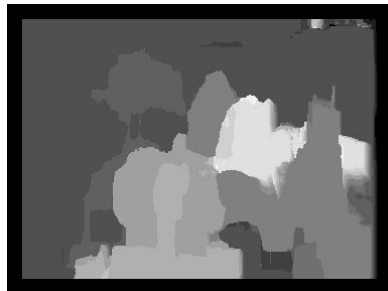
$ws_{\text{ize}} = 21 \times 21$

Global Error = 12.2604

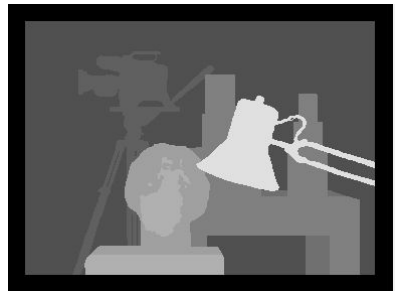


$ws_{\text{ize}} = 31 \times 31$

Global Error = 13.4334



Ground truth



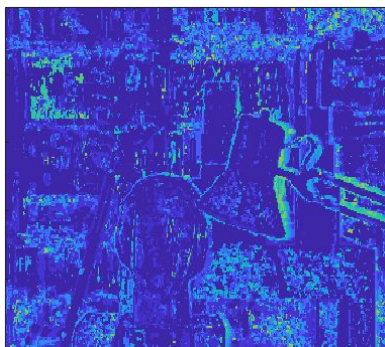
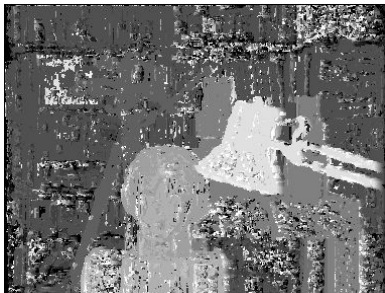
Question 5. Complete the 'stereo_computation' function with the computation of the NCC cost.

```
elseif strcmpi(cost_function, 'NCC') %Normalized Cross Correlation
    error = -Inf;
    for k = kmin:kmax
        windowl = double(Ir(i-w_step:i+w_step, k-w_step:k+w_step,:));
        term1 = windowl-sum(sum(sum(weight.*windowl)));
        term2 = windowr-sum(sum(sum(weight.*windowr)));
        sigma1l = sqrt(sum(sum(sum(weight.*term1.^2))));
        sigma1r = sqrt(sum(sum(sum(weight.*term2.^2))));
        costNCC = sum(sum(sum(weight.*term1.*term2)))/(sigma1l*sigma1r);

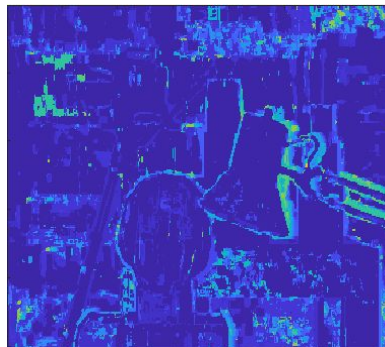
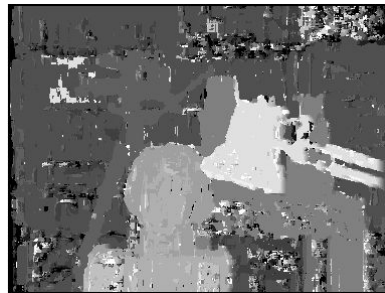
        if(costNCC>error)
            error = costNCC;
            kbest = k;
        end
    end
end
```

Question 6. Compare the results of NCC with those obtained with the SSD cost.

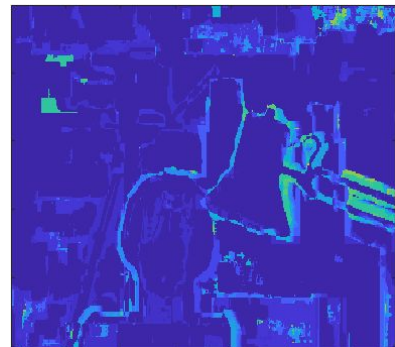
wsizer = 3
Global Error = 24.7589



wsizer = 5
Global Error = 17.0210



wsizer = 9
Global Error = 12.7504



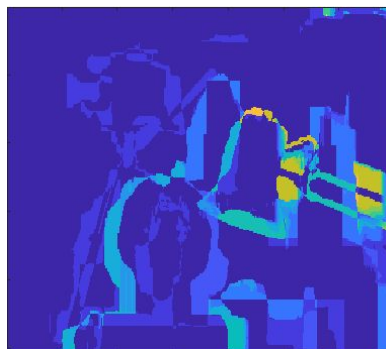
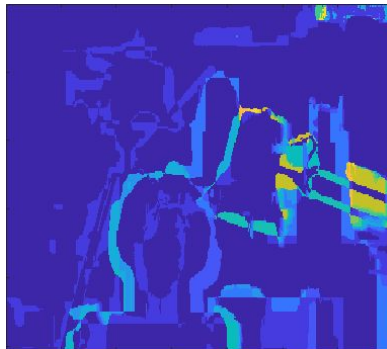
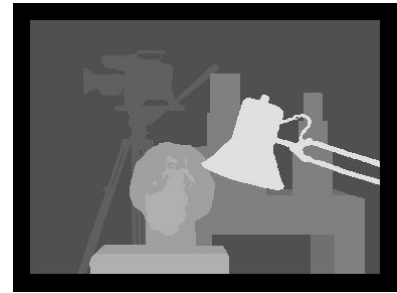
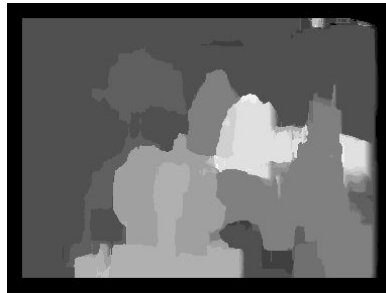
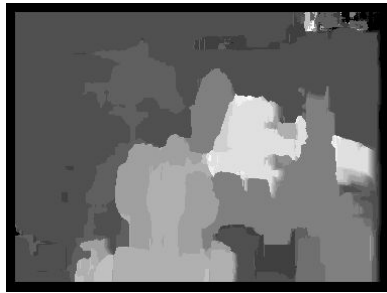
wsizer = 21

Global Error = 12.2190

wsizer = 31

Global Error = 13.5283

Ground truth



En termes de *Global Error*, utilitzant NCC s'obtenen uns valors amb una variança superior a quan s'utilitza SSD. Tot i així, el mínim error global que hem trobat fins ara és 12.2190 i ha sigut utilitzant NCC amb una finestra de 21x21.

Question 7. Complete the 'stereo_computation' function with the bilateral weights

```
% Bilateral Weights
if(bilateral)
    % TO COMPLETE

    Np = sum(window1,3);
    P = zeros(wsize);
    centerPoint = floor(wsize/2);
    P(:, :) = Np(centerPoint,centerPoint);
    C = (1/3)*abs(P-Np);
    wcol = exp(-C/stdC);

    Q = [];
    q = [];

    for n = i-w_step:i+w_step
        for m = j-w_step:j+w_step
            q = [n,m];
            Q = [Q;q];
        end
    end

    end
end

G = sqrt(sum((Q-Q).^2,2));
wpos = exp(-G/stdP);
wpos = (reshape(wpos,[wsize,wsize]))';

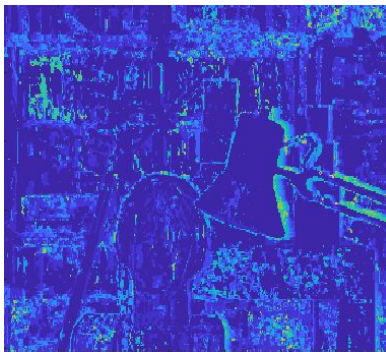
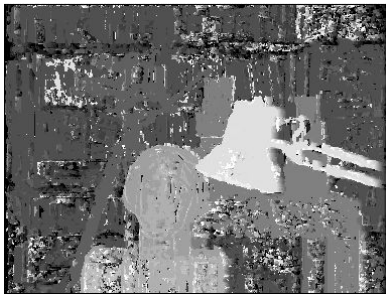
weight = wcol.*wpos;

end
```

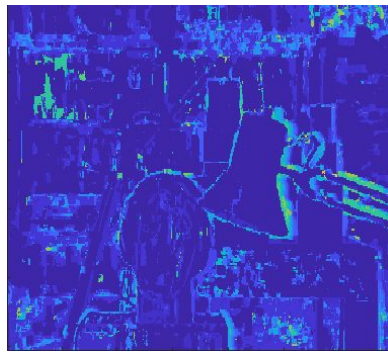
Question 8. Evaluate the results changing the window size (e.g. 5x5, 9x9, 31x31) and compare to the previous case that uses uniform weights (SSD cost).

Els resultats que veurem a continuació són obtinguts utilitzant simultàniament SSD cost, i bilateral weights per a obtenir uns pesos dinàmics:

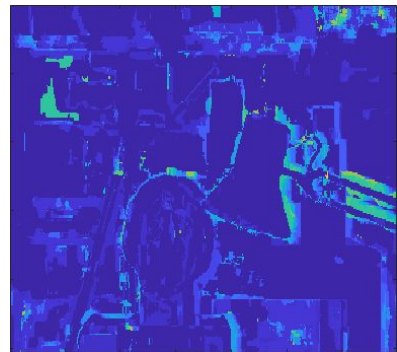
$ws_{size} = 3$
Global Error = 21.2448



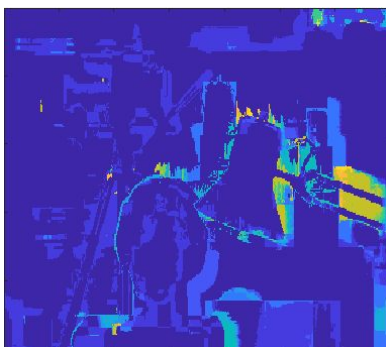
$ws_{size} = 5$
Global Error = 16.2103



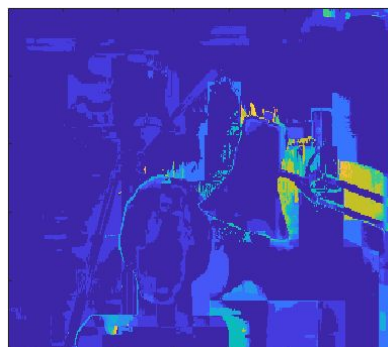
$ws_{size} = 9$
Global Error = 13.1138



$ws_{size} = 21$
Global Error = 11.1426



$ws_{size} = 31$
Global Error = 11.3600



Ground truth



Com podem veure, tot i que el resultat no és ben bé l'esperat, l'error ha disminuït i es pot veure clarament l'efecte del bilateral weight ja que el resultat és molt més precís.

Question 9. Comment the result obtained. Why do you think the disparity is not well estimated in parts of the table?

En aquesta última pregunta hem utilitzat les imatges de la pràctica anterior. La raó per la qual la disparitat no ha sigut ben estimada en algunes parts de la taula és degut a la seva homogeneïtat.

