
```

function sim = compute_similarity(dets_cur, dets_next, im_cur,
    im_next)

n = size(dets_cur, 1);
m = size(dets_next, 1);
sim = zeros(n, m);

area_cur = compute_area(dets_cur);
area_next = compute_area(dets_next);
c_cur = compute_center(dets_cur);
c_next = compute_center(dets_next);
im_cur = double(im_cur);
im_next = double(im_next);
weights = [1,1,2];

for i = 1: n
    % compare sizes of boxes
    a = area_cur(i) * ones(m, 1);
    sim(i, :) = sim(i, :) + weights(1) * (min(area_next, a) ./
max(area_next, a))';

    % penalize distance (would be good to look-up flow, but it's slow
to
    % compute for images of this size)
    sim(i, :) = sim(i, :) + weights(2) *
exp((-0.5*sum((repmat(c_cur(i, :), [size(c_next, 1), 1]) -
c_next).^2, 2)) / 5^2))';

    % compute similarity of patches
    box = round(dets_cur(i, 1:4));
    box(1:2) = max([1,1],box(1:2));
    box(3:4) = [min(box(3),size(im_cur, 2)), min(box(4),size(im_cur,
1))];
    im_i = im_cur(box(2):box(4),box(1):box(3), :);
    im_i = im_i / norm(im_i(:));
    for j = 1 : m
        d = norm(c_cur(i, :) - c_next(j, :));
        if d>60 % distance between boxes too big
            sim(i,j) = 0;
            continue;
        end;
        box = round(dets_next(j, 1:4));
        box(1:2) = max([1,1],box(1:2));
        box(3:4) = [min(box(3),size(im_cur, 2)),
min(box(4),size(im_cur, 1))];
        im_j = im_next(box(2):box(4),box(1):box(3), :);
        im_j = double(imresize(uint8(im_j), [size(im_i, 1), size(im_i,
2)]));
        im_j = im_j / norm(im_j(:));
        c = sum(im_i(:) .* im_j(:));
        sim(i,j) = sim(i,j) + weights(3) * c;
    end
end

```

```
    end;  
end;  
end
```

Not enough input arguments.

*Error in compute_similarity (line 3)
n = size(dets_cur, 1);*

Published with MATLAB® R2017b