%% Mean-Shift Video Tracking

% by Sylvain Bernhardt

% July 2008

%% Description

% This is a simple example of how to use

% the Mean-Shift video tracking algorithm

% implemented in 'MeanShift\_Algorithm.m'.

% It imports the video 'Ball.avi' from

% the 'Videos' folder and tracks a selected

% feature in it.

% The resulting video sequence is played after

% tracking, but is also exported as a AVI file

% 'Movie\_out.avi' in the 'Videos' folder.

clear all

close all

%% Import movie and time it with tic/toc

tic

[Length,height,width,Movie]=Import\_mov('Videos/ball.avi');

toc

%% Variables

index\_start = 1;%videonun frame nosunu tutacak indis

% mean shift convergence benzerlik eşiği, bu değerin altındaysa hedef bulundu

f\_thresh = 0.16;%f\_thresh = 0.16;

% mean shift conv. itr. sayısı, her framede en fazla bu itr. kadar ort. hesaplanacak

max\_it = 15;%pembe için 15, yeşil 30, mavi 5

% seçilen bölge kadar boyutlu kernel penceresi(her noktanın ağırlığını belirleyecek)

kernel\_type = 'Epanechnikov';%kernel\_type = 'Epanechnikov';

radius = 2;%radius = 2;

%% Target Selection in Reference Frame

%ilk framede takip edilecek objeyi seçme

%seçilen bölge(T), sol üst köşesi(x0,y0), yüksekliği(H) ve genişliği(W)

[T,x0,y0,H,W] = Select\_patch(Movie(index\_start).cdata,0);

pause(0.2);

%% Run the Mean-Shift algorithm

% Calculation of the Parzen Kernel window

%seçilen büyüklükte parzen window, gauss dağılımından ağırlıklar

%seçilen alanın merkezindeki piksellerin ağırlığı daha yüksek olsun objeyi

%temsil etme gücü yüksek olan birimler

%ağırlık penceresi(k), x yönündeki gradyeni(gx), y yönündeki gradyeni(gy)

[k,gx,gy] = Parzen\_window(H,W,radius,kernel\_type,0);

% Conversion from RGB to Indexed colours

% to compute the colour probability functions (PDFs)

%her renk tonuna bir index atanarak görüntüler index numaraları şeklinde

%ifade ediliyor

[I,map] = rgb2ind(Movie(index\_start).cdata,65536);

Lmap = length(map)+1;%kaç farklı ton var

T = rgb2ind(T,map);%seçilen bölge de ilk görüntüde belirlenen indexler şeklinde ifade ediliyor

% Estimation of the target PDF

% seçilen bölgedeki her bir renk tonunun konumsal ağırlığına bağlı olarak

% histogramın(q) hesaplanması

q = Density\_estim(T,Lmap,k,H,W,0);

% Flag for target loss

loss = 0;%hedefi kaybetti mi?

% Similarity evolution along tracking

f = zeros(1,(Length-1)\*max\_it);%her framede max\_it sayısı kadar hesaplanan benzerlik değerleri

% Sum of iterations along tracking and index of f

f\_indx = 1;%kaçıncı benzerlik yazılacak?

% Draw the selected target in the first frame

%hedefe ilk frame üzerinde çerçeve çizilmesi

Movie(index\_start).cdata = Draw\_target(x0,y0,W,H,...

Movie(index\_start).cdata,2);

%% TRACKING

WaitBar = waitbar(0,'Tracking in progress, be patient...');

% From 1st frame to last one

%videonun sonuna kadar her frame için mean shift uygulanır

for t=1:Length-1

% Next frame

I2 = rgb2ind(Movie(t+1).cdata,map);%sıradaki frame'i ilk framede belirlediğimiz renk haritasındaki indislerle ifade ediyoruz

% Apply the Mean-Shift algorithm to move (x,y)

% to the target location in the next frame.

%sıradaki framede nesnenin yerini(x,y) bulmak için mean shift uygulanır

%I2 frame'inde q histogramına en benzer çerçeveyi bulmak

[x,y,loss,f,f\_indx,w] = MeanShift\_Tracking(q,I2,Lmap,...

height,width,f\_thresh,max\_it,x0,y0,H,W,k,gx,...

gy,f,f\_indx,loss);

% Check for target loss. If true, end the tracking

if loss == 1

break;

else

% Drawing the target location in the next frame

%bir sonraki framede hedef üzerinde çerçeve çizilir

Movie(t+1).cdata = Draw\_target(x,y,W,H,Movie(t+1).cdata,2);

% Next frame becomes current frame

y0 = y;

x0 = x;

% Updating the waitbar

waitbar(t/(Length-1));

end

end

close(WaitBar);

%% End of TRACKING

% güncellenmiş videoyu göster

scrsz = get(0,'ScreenSize');

figure(1)

set(1,'Name','Movie Player','Position',...

[scrsz(3)/2-width/2 scrsz(4)/2-height/2 width height],...

'MenuBar','none');

axis off

% Image position inside the figure

set(gca,'Units','pixels','Position',[1 1 width height])

% Play the movie

movie(Movie);

%% End of File

%=============%

%% Mean-Shift Video Tracking

% by Sylvain Bernhardt

% July 2008

% Updated March 2012

function [lngth,h,w,mov]=Import\_mov(path)

infomov = VideoReader(path);

lngth = infomov.numberOfFrames;%video kaç frameden oluşuyor?

h = infomov.Height;%videonun bir frameinin yüksekliği

w = infomov.Width;%videonun bir frameinin genişliği

mov(1:lngth) = struct('cdata', zeros(h, w, 3, 'uint8'), 'colormap', []);

% her bir frame mov yapısının bir elemanı olacak

for k = 1 : lngth

mov(k).cdata = read(infomov, k);

end

%% Mean-Shift Video Tracking

% by Sylvain Bernhardt

% July 2008

%% Description

% It permit to select a patch T from

% an image I.

% The outputs are the patch T, its upper left

% corner location x0,y0 and size H,W.

% If 'graph' is set to 1, it also displays

% the selected patch in a new figure.

%

% [T,x0,y0,H,W]=Select\_patch(I,graph)

function [T,x0,y0,H,W]=Select\_patch(I,graph)

height = size(I,1);

width = size(I,2);

% Put the figure in the center of the screen,

% without menu bar and axes.

scrsz = get(0,'ScreenSize');

figure (2)

set(2,'Name','Target Selection','Position',...

[scrsz(3)/2-width/2 scrsz(4)/2-height/2 width height],...

'MenuBar','none');

axis off

% Image position inside the figure

% görüntü pencerenin sol alt köşesine hizalansın

set(gca,'Units','pixels','Position',[1 1 width height])

% Display the image

imshow(I);

rect = getrect;%mouse ile çizilen dikdörtgeni al

rect = floor(rect);

%çizilen dikdörtgenin başlangıç noktası (x0,y0) genişliği(W) ve uzunluğu(H)

x0 = rect(1);

y0 = rect(2);

W = rect(3);

H = rect(4);

T = I(y0:y0+H-1,x0:x0+W-1,:);%dikdörtgen içinde kalan kısımdaki görüntü(T)

if graph==1

figure (3)

set(3,'Name','Target Selected')

imshow(T);

end

close (2)

%% Mean-Shift Algorithm

% by Sylvain Bernhardt

% July 2008

%% Description

% Computes the mask of a Parzen window

% and its gradient in respect of the x-axis

% and the y-axis.

% The different types of kernel are:

% {Uniform,Triangular,Epanechnikov,Gaussian}

%

% [k,gx,gy] = Parzen\_window(H,W,R,type,graph)

% with:

% k - the mask

% gx,gy - its gradients

% (H,W) - size of the mask

% type - its type

% graph - plot the masks if graph=1

function [k,gx,gy] = Parzen\_window(H,W,R,type,graph)

k = zeros(H,W);

%% ----Uniform----

if strcmp(type,'Uniform')==1

for i=1:H

for j=1:W

if (((2\*i)/H-1)/R)^2+(((2\*j)/W-1)/R)^2 <= 1

k(i,j) = 1;

end

end

end

end

%% ----Triangular----

if strcmp(type,'Triangular')==1

Max = max(H,W);

for z=1:round(R\*Max/2)-1

h = zeros(H,W);

for i=1:H

for j=1:W

if ((i-(H/2))/(R\*H/2-z\*H/Max))^2+...

((j-(W/2))/(R\*W/2-z\*W/Max))^2 <= 1

h(i,j) = 2/(Max\*R);

end

end

end

k = k+h;

end

end

%% ----Epanechnikov

if strcmp(type,'Epanechnikov')==1

for i=1:H

for j=1:W

k(i,j) = (1-(2\*i/(R\*H)-1/R)^2-...

(2\*j/(R\*W)-1/R)^2);

if k(i,j) < 0

k(i,j) = 0;

end

end

end

end

%% ----Gaussian----

if strcmp(type,'Gaussian')==1

%gauss fonk. x ve y yönündeki standart sapmaları

sigmaH = (R\*H/2)/3;

sigmaW = (R\*W/2)/3;

% sigma = x/3 as a gaussian is almost equal to 0

% from 3\*sigma.

for i=1:H

for j=1:W

k(i,j) = exp(-.5\*((i-.5\*H)^2/sigmaH^2+...

(j-.5\*W)^2/sigmaW^2));

end

end

end

%% Gradient of kernel

[gx,gy] = gradient(-k);

%% Plotting the window

if graph==1

figure (4)

scrsz = get(0,'ScreenSize');

set(4,'Position',[scrsz(3)/4 scrsz(4)/4 ...

scrsz(3)/1.5 scrsz(4)/1.5])

subplot(2,2,1)

mesh(k);

surf(k);

shading interp

subplot(2,2,2)

mesh(gx);

surf(gx);

shading interp

subplot(2,2,3)

mesh(gy);

surf(gy);

shading interp

end

%% Mean-Shift Video Tracking

% by Sylvain Bernhardt

% July 2008

%% Description

% Estimate the density of data samples

% (here colour histogram) in a patch T

% with a kernel profile k. Lmap is the

% colormap length and H,W the patch size.

function q = Density\_estim(T,Lmap,k,H,W,graph)

q = zeros(Lmap,1);

colour = linspace(1,Lmap,Lmap);%1'den ton sayısına kadar her elemanı tonun indexi olan dizi

for x=1:W

for y=1:H

%aynı tonda olan piksellerin ağırlıkları kendi indexinde toplanıyor

q(T(y,x)+1) = q(T(y,x)+1)+k(y,x);

end

end

% Normalizing

C = 1/sum(sum(k));%normalizasyon katsayısı, histogram toplamı 1'e eşit olacak şekilde normalize edilecek

q = C.\*q;%normalize edilmiş renk histogramı

% Plotting the estimated densities

if graph==1

figure

plot(colour,q);

end

%% Mean-Shift Video Tracking

% by Sylvain Bernhardt

% July 2008

%% Description

% Insert in the image I a rectangle

% which size is H,W and thickness is

% expressed in pixels.

%

% I = Draw\_target(x,y,W,H,I,thick)

function I = Draw\_target(x,y,W,H,I,thick)

R=1; G=2; B=3;

I(y:y+thick-1,x:x+W,R) = 255;%üst

I(y:y+thick-1,x:x+W,G) = 0;

I(y:y+thick-1,x:x+W,B) = 0;

I(y+H-thick+1:y+H,x:x+W,R) = 255;%alt

I(y+H-thick+1:y+H,x:x+W,G) = 0;

I(y+H-thick+1:y+H,x:x+W,B) = 0;

I(y:y+H,x:x+thick-1,R) = 255;%sağ

I(y:y+H,x:x+thick-1,G) = 0;

I(y:y+H,x:x+thick-1,B) = 0;

I(y:y+H,x+W-thick+1:x+W,R) = 255;%sol

I(y:y+H,x+W-thick+1:x+W,G) = 0;

I(y:y+H,x+W-thick+1:x+W,B) = 0;

%% Mean-Shift Video Tracking

% by Sylvain Bernhardt

% July 2008

%% Description

% Tracks a patch 'T' in a video sequence 'Movie'

% using the Mean-Shift algorithm.

% f is the similiraty function between the original patch(q)

% and the candidate ones(p) along the video sequence.

%

% [x,y,loss,f,f\_indx] = MeanShift\_Tracking(q,I2,Lmap,...

% height,width,f\_thresh,max\_it,x0,y0,H,W,k,gx,gy,...

% f,f\_indx,loss)

% with:

% (x,y) - the location of the target in the frame I2

% (f,f\_indx) - storing the evolution of similarity

% loss - flag for target loss

% q - PDF of the reference target

% I2 - the next frame

% Lmap - length of colormap, also number of bins for PDF

% height,width - size of I2

% f\_thresh - the similarity threshold

% max\_it - the maximum number of iterations

% x0,y0 - the location of the target

% H,W - its size

% (k,gx,gy) - kernel mask and its gradients

function [x,y,loss,f,f\_indx,w] = MeanShift\_Tracking(q,I2,Lmap,...

height,width,f\_thresh,max\_it,x0,y0,H,W,k,gx,gy,f,f\_indx,...

loss)

% Initialization in the next frame from the

% same location than in the current frame.

%ilk aday-> ilk framede seçilen bölgenin sonraki framede aynı

%koordinatlardaki histogramı

y = y0;

x = x0;

T2 = I2(y:y+H-1,x:x+W-1);%seçilen koordinatlardaki 2.framedeki görüntü parçası(T2)

p = Density\_estim(T2,Lmap,k,H,W,0);%adayın histogramı

% Number of iterations

step = 1;

% Computation of the similarity value

% between the two PDF.

[fi,w] = Simil\_func(q,p,T2,k,H,W);%iki histogram arasındaki benzerlik

f = cat(2,f,fi);%bulunan değer benzerlik vektörüne ekleniyor

% Applying Mean-shift algorithm

%yakınsama koşulları 1. -> hist. arası farkın f\_threshden küçük olması

%2. -> max\_it iterasyon sınırına ulaşmış olmak(her framede en fazla kayma itr.)

while f(f\_indx)<f\_thresh && step<max\_it %max\_it=5 iken 4 kayma yapabilir.

step = step+1;

f\_indx = f\_indx+1;

num\_x = 0;

num\_y = 0;

den = 0;

%new target center'ın belirlenmesi

for i=1:H

for j=1:W

%her pikselin x ve y yönündeki gauss gradyenleri ile çarpılır

num\_x = num\_x+i\*w(i,j)\*gx(i,j);

num\_y = num\_y+j\*w(i,j)\*gy(i,j);

den = den+w(i,j)\*norm([gx(i,j) gy(i,j)]);

end

end

% Displacement vector (dx,dy) on the gradient ascent

%gradyenin en yüksek olduğu yere doğru mean shift vektör oluşturulur(dx,dy, x ve y yönündeki bileşenleri)

%merkezin vektör doğrultusunda güncellenmesi

if den ~= 0

dx = round(num\_x/den);

dy = round(num\_y/den);

y = y+dy;

x = x+dx;

end

% Detection of target loss or out of frame boundaries

if (y<1 || y>height-H) || (x<1 || x>width-W)

loss = 1;

Target\_Loss\_Dialog\_Box();%çizilen bölge video görüntüsünden çıktıysa

uiwait(Target\_Loss\_Dialog\_Box);

break;

end

% 2. iterasyon için Update the target

T2 = I2(y:y+H-1,x:x+W-1);

p = Density\_estim(T2,Lmap,k,H,W,0);

[fi,w] = Simil\_func(q,p,T2,k,H,W);

f = cat(2,f,fi);

end

%% Mean-Shift Video Tracking

% by Sylvain Bernhardt

% July 2008

%% Description

% Measures the similarity between two

% density estimations q and p done with

% a kernel which profile is k.

% q is the estimation of a reference patch

% and p the estimation of a candidate one 'T2'

% which size is H,W.

% The outputs are the similarity value f

% and the weight mask w for the gradient ascent

% in the extended Mean-Shift algorithm.

%

% [f,w] = Simil\_func(q,p,T2,k,H,W)

function [f,w] = Simil\_func(q,p,T2,k,H,W)

w = zeros(H,W);

f = 0;

for i=1:H

for j=1:W

%benzerliği artırmak için taylor açılımındaki ağırlık ifadesinin bulunması

%pikselin histogramlardaki bin'lerinin birbirine bölümü

w(i,j) = sqrt(q(T2(i,j)+1)/p(T2(i,j)+1));%maximizing bhattacharyya coefficient

%pikselin gauss ağırlığı ile çarpılıp tüm pikseller toplanır

f = f+w(i,j)\*k(i,j);

end

end

% Normalization of f

f = f/(H\*W);