

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΑΤΡΩΝ

ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ : Τ.Μ.Η/Υ.Π

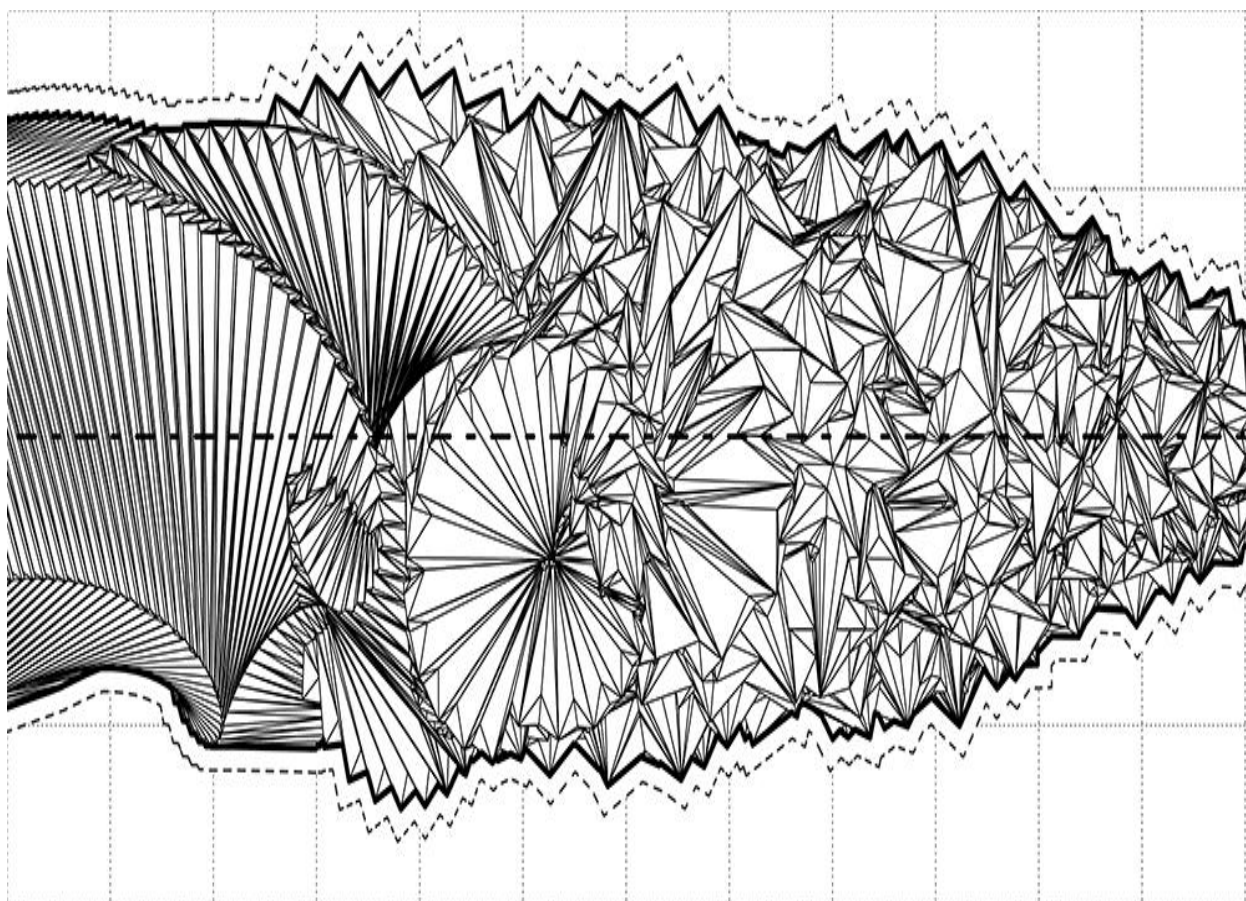
ΕΠΩΝΥΜΟ : ΤΡΙΑΝΤΗΣ

ΟΝΟΜΑ : ΠΑΝΑΓΙΩΤΗΣ

Α.Μ : 5442

ΥΠΟΛΟΓΙΣΤΙΚΗ ΟΡΑΣΗ & ΓΡΑΦΙΚΗ

ΒΑΣΙΚΟΙ ΓΕΩΜΕΤΡΙΚΟΙ ΜΕΤΑΣΧΗΜΑΤΙΣΜΟΙ



ΔΙΑΔΙΚΑΣΙΑ

1)

- **imread** : Είναι η εντολή που «διαβάζει» τις εικόνες και τις μετατρέπει σε ένα μητρώο στη matlab. Έχει διάφορες λειτουργίες όπως να γεμίζει το background της εικόνας με άσπρο,μαύρο, να επιλέγει από κάποια εικόνα συγκεκριμένο εύρος pixel που θέλουμε να διαβάσει και να τα αποθηκεύσει σε ένα μητρώο.
- **imwarp** : Η συγκεκριμένη εντολή εφαρμόζει έναν γεωμετρικό μετασχηματισμό στην εικόνα που θέλουμε. Μπορούμε να χρησιμοποιήσουμε διάφορες λειτουργίες όπως τη 'FillValues' που μετά το μετασχηματισμό γεμίζει τα pixel που είναι εκτός ορίων της εικόνας με τη τιμή που θέλουμε, την 'OutputView' που επιλέγουμε την θέση της εξαγόμενης εικόνας μέσα σε ένα σύμπαν που έχουμε ορίσει. Ακόμη έχει λειτουργίες που καθορίζουμε το τύπο της παρεμβολής που θα χρησιμοποιήσει (nearest,linear,cubic).
- **affine2d** : Δημιουργεί το μητρώο affine(tform) που θέλουμε για να κάνουμε το γεωμετρικό μετασχηματισμό της εικόνας. Δέχεται το μητρώο affine που θέλουμε μόνο που για το translation το πόσο θα μετακινηθεί η εικόνα και προς τα πού ορίζεται στη τελευταία γραμμή του μητρώου. Π.χ

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ tr_x & tr_y & 1 \end{bmatrix}$$

- **projective2d** : Παρόμοια και η projective2d όπως και η affine2d δημιουργεί ένα tform το οποίο χρησιμοποιείται από την imwarp για να μετασχηματίσει την εικόνα.
- **imref2d** : Γίνεται αναφορά μιας εικόνας σε διδιάστατες συντεταγμένες που επιλέγουμε. Αυτή η εντολή είναι χρήσιμη για την imwarp ώστε να χρησιμοποιήσουμε τις συντεταγμένες που θέλουμε για να τοποθετήσουμε την εικόνα που μετασχηματίζουμε στο επιθυμητό σημείο (translation).
- **implay** : Εντολή της matlab που παίζει βίντεο, ταινίες ή ακολουθίες εικόνων (frames). Μπορεί να επιλεγεί η ταχύτητα -to framerate- (fps) του επιλεγμένου αρχείου που θέλουμε.

2) Στα πλαίσια αυτού του ερωτήματος δημιουργήθηκε το script **scaling_picture.m** . Στα σχόλια αναφέρονται όλες οι επιλογές και πως δημιουργήθηκε η εικόνα που ζητείται. Εδώ παρουσιάζω την αρχική εικόνα και αυτή που δημιουργήθηκε ύστερα από το μετασχηματισμό.



Για τα ερωτήματα 3,4,5 παραθέτω τους κώδικες οι οποίοι είναι σχολιασμένοι και απαντούν σε ότι ζητείται. Τα αρχεία που ζητούνται βρίσκονται σε φάκελο όλα μαζί και ανεβασμένα στο dropbox στο link που δίνω. Μπορείτε να τα κατεβάσετε ή να τρέξετε τους κώδικες τα αποτελέσματα δεν αργούν πολύ (εδώ να αναφέρω ότι χρησιμοποιώ τη Matlab R2013b).

<https://www.dropbox.com/l/sh/kzvGxF2gVVIAlYQM4se9Pp>

Ακόμη να σχολιάσω για το ερώτημα 4, για την επιλογή της παρεμβολής κατά το μετασχηματισμό της εικόνας :

- **linear** : είναι η default παρεμβολή για τη συνάρτηση `imwarp` της matlab. Το αποτέλεσμα με αυτού του είδους τη παρεμβολή είναι ικανοποιητικό από άποψη ποιότητας και επεξεργασίας.
- **nearest** : είναι η παρεμβολή που επιλέγει το κοντινότερο σημείο από το άγνωστο σημείο της παρεμβολής. Από άποψη επεξεργασίας χρειάζεται λιγότερη από τις παρεμβολές των δύο άλλων ειδών αλλά από άποψη ποιότητας είναι φανερά χαμηλότερη των άλλων. Αυτό είναι αρκετά εμφανές στο βίντεο μας όταν χρησιμοποιείται αυτού του είδους παρεμβολή.
- **cubic** : η cubic έχει καλύτερα αποτελέσματα και από τις δύο παρεμβολές πιο πάνω σε άποψη ποιότητας αλλά η επεξεργασία που χρειάζεται είναι αρκετά μεγαλύτερη.

Οι κώδικες που χρησιμοποιήθηκαν στα πλαίσια της ασκήσεως

```
%% script gia thn eikona pou zhth8hke se diafores klimakes

close all; clear all; clc

moon = im2double(imread('redmoon.jpg'));
red_moon = imresize(moon,[375 500]);

backgr = zeros(2*size(red_moon,1),2*size(red_moon,2),3);
rout = imref2d(size(backgr));
tx = min(rout.XWorldLimits);
```

```

ty = min(rout.YWorldLimits);
tx1 = mean(rout.XWorldLimits);
ty1 = mean(rout.YWorldLimits);
counter = 0;

% edw ginetai h klimakwh ths eikonas (th mikrainw) kai topo8eteitai
% mesa se
% ena mauro plaio apo thn imwarp ka8ws xrhsimopoiw to imref2d kai
% kanw
% translation ta apotelesmata mesa se ena "background" poy exei to
% mege8os
% pou exw epile3ei sto imref2d
for i = 1:6
    if i<=1
        j = 0;
    elseif i>1
        counter = counter+1;
        j = 1*counter;
    end
    a = [1/i 0 0;
         0 1/i 0;
         tx1-tx1/i ty1-ty1*j/i 1];
    aff = affine2d(a);
    im = imwarp(red_moon,aff,'OutputView',rout);
    img_n{i,1} = im;
end

% edw ginetai mia aplh enwsh tw'n apotelesmatwn pou epitygxanetai me
% pros8esh tw'n eikonwn
scaling = img_n{1}+img_n{2}+img_n{3}+img_n{4}+img_n{5}+img_n{6};

% edw gia na er8w se ena apotelesma pio wraio kanw to e3hs
% arxika me to transpose gyrizw thn eikona kai th kanw meta ena
% rotation
% se oles tis diastaseis ths kai th pros8etw me thn eikona apo pio
% panw
for i = 1:3
    scale(:, :, i) = scaling(:, :, i)';
end
scale_tel = scaling + imrotate(scale,-90);
figure,imshow(scale_tel)

% me thn imwrite egine h apo8hkeush ths eikonas
% imwrite(scale_tel,'feggari_scale.jpg')

```

```

%% pudding motion script

```

```

close all; clear all; clc

```

```

img = imread('pudding.png','BackgroundColor',[1;1;1]);

```

```

%gyrname anapoda tis eikones giati an tis metasxhmatismo me tis
%eikones
%kanonika o patos ths poutigkas gyrizei kai de menei sta8eros logw
%tou
%mege8ous ths
for j = 1:3
    im(:, :, j) = flipud(img(:, :, j));

```

```

end

%ftiaxnoume 2 cell gia na apo9hkeusoume ta frames pou apaitountai gia
na
%dhmiourgh8ei h kinhsh pou 8eloume
%sto gyrth1 apo8hkeuoume thn kinhsh pros ta de3ia kai sto gyrth2
%apo8hkeuoume th kinhsh pros ta aristera
gyrth1 = cell([25,1]);
gyrth2 = cell([25,1]);

imm = ones(512,512,3);
rout = imref2d(size(imm));

for i1 = 0:24
    x = i1*0.01;
    a = [1 0 0;
         x 1 0;
         125 125 1;];
    metasx = affine2d(a);
    gyr = imwarp(im,metasx,'FillValues',255,'OutputView',rout);
    gyrth1{i1+1,1} = gyr;
end
%edw afoy exoume apo8hkeusei th kinhsh pros ta de3ia sto gyrth1
%dhmiourgoume ena allo cell , to gyrth1_2 to opoio einai h kinhsh pou
%periexei to gyrth1 alla anti8eta, dhladh apo de3ia mexri thn arxikh
8esh
gyrth1_2 = flipud(gyrth1);

%h idia diadikasia epanalamvanetai kai gia th kinhsh pros ta aristera
opws
%sto gyrth1
for i2 = 0:24
    x2 = i2*(-0.01);
    a2 = [1 0 0;
          x2 1 0;
          125 125 1;];
    metasx2 = affine2d(a2);
    gyr2 = imwarp(im,metasx2,'FillValues',255,'OutputView',rout);
    gyrth2{i2+1,1} = gyr2;
end
gyrth2_2 = flipud(gyrth2);

%dhmiourgia enos cell pou 8a periexei ola ta frames
f1 = cell([100,1]);

%edw apo8hkeuoume sto cell f1 ola ta frames pou dhmiourghsame
(synolika ta
%frames einai (25 + 25)*2 ara 100)
%apo 1:50 ta frames einai kinhshs apo th kanonikh 8esh ews de3ia kai
pali pishw
%enw 50:100 ths aristerhs ews to kentro kai pali pishw

for i = 1:100
    if i <=25
        f1{i,1} = gyrth1{i,1};
    end
    if i>=26 && i<=50
        f1{i,1} = gyrth1_2{i-25,1};
    end
end

```

```

    if i>=51 && i<=75
        f1{i,1} = gyrth2{i-50,1};
    end
    if i>=76 && i<= 100
        f1{i,1} = gyrth2_2{i-75,1};
    end
    f1{i,1} = imresize(f1{i,1},[256 256]);
end

%pleon anapodogyrizontai ola ta frames outws wste h eikona na parei
th
%kanonikh ths morfhs
for jk = 1:100
    for jj = 1:3
        f1{jk,1}(:, :, jj) = flipud(f1{jk,1}(:, :, jj));
    end
end

%telika dhmiogyrizontai ena struct me thn im2frame pou metatrepei tis
% parapanw eikones se frames
for vid = 1:100
    f(vid) = im2frame(f1{vid,1});
end

%to teliko video
imshow(f,25)



---



%% windmill motion script

close all; clear all; clc

ftr = rgb2gray(im2double(imread('windmill.png')));
windm_back = rgb2gray(im2double(imread('windmill_back.jpeg')));
mask = rgb2gray(im2double(imread('windmill_mask.png')));

cmap = gray(256);
ftrh = cell([200,1]);
rout = imref2d(size(windm_back));
tx = mean(rout.XWorldLimits);
ty = mean(rout.YWorldLimits);
ftr2 = (1-ftr).*(1-mask);
%metasxhmatismos arxikos
% edw pairnoume thn eikona apo thn elika tou anemomylou kai th
bazoume na
% kineitai kyklika (rotation)
for i = 0:199
    a = [cos((pi/90)*i) sin((pi/90)*i) 0;
        -sin((pi/90)*i) cos((pi/90)*i) 0;
        0 0 1];
    metasx = affine2d(a);
    gyr = imwarp(ftr2,metasx,'Interpolation','linear');
    ftrh{i+1,1} = gyr;
end
% kanonikopoioume ta mege8h twv eikonwn meta to metasxhmatismo kai
auto
% giati oi eikones pou dhmiourgountai logw tou rotation, prokypoun k
exoun

```

```

% diafora mege8h. auto symvainei giati megalwnoun ta boundaries ths
eikonas
% otan peristrefetai
for ii = 1:200
    [x,y,~] = size(fterh{ii,1});
    if x-512>0 % ousiastika elegxw poies eikones exoun mege8os
megalytero
        cr = (x-512)/2; % apo thn arxikh kai epilegw ta stoixeia pou
eixe
        cr = floor(cr); % h arxikh eikona thn kanw peripoy sto arxiko
mege8os
        fterh{ii,1} = fterh{ii,1}((cr+1):x-(cr+1),(cr+1):x-(cr+1),:);
    elseif x-512==0
        fterh{ii,1} = fterh{ii,1};
    end
end
%tis kainouries pleon eikones tis metasxhmatizoume se ena background
iso me
%to size tou background mas kai tis topo8etoume sto shmeio
peristrofhs me
%th xrhsh toy rout mesa sthn "imwarp"
for i = 1:200
    i2 = i/2;
    a = [1 0 0;
        0 1 0;
        tx/2+50 ty/2-70 1];
    metasx2 = affine2d(a);
    gyr2 =
imwarp(fterh{i,1},metasx2,'OutputView',rout,'Interpolation','linear')
;
    fterh{i,1} = gyr2;
end
%syn8etoume tis eikones pou 8a ginoun frames pleon kai trexoume to
implay
for i = 1:200
    fterh{i,1} = (1-fterh{i,1}(:,:,:)).*windm_back(:,:,:);
    %edw sto wndm mesa sto imframe metasxhmatizoume tis eikones pali
se
    %uint8 giati alliws de mporousame na xrhsimopoihsoume thn entolh
%im2frame kai bazoume san colormap to cmap pou to exoume 8esei
sthn
    %arxh
    wndm(i) = im2frame(uint8(round(255*fterh{i,1})),cmap);
end
implay(wndm,20)

```

```

%% script beach_motion

```

```

close all; clear all; clc

```

```

ball = rgb2gray(im2double(imread('ball.jpg')));
ball_mask = rgb2gray(im2double(imread('ball_mask.jpg')));
beach = rgb2gray(im2double(imread('beach.jpg')));

```

```

cmap = gray(256);

```

```

ball = imresize(ball,0.125);
ball_mask = imresize(ball_mask,0.125);
beach = imresize(beach,[600 1280]);

```

```

rout = imref2d(size(beach));
tx = mean(rout.XWorldLimits);
ty = mean(rout.YWorldLimits);

%ball1 = (1-ball_mask).*(1-ball);
ball1 = 1-ball;
spin = cell([200,1]);
beach_vid = cell([200,1]);

%prwta ginetai o metasxhmatismos pou kanei th mpala na peristrefetai
for i = 0:199
    a = [cos((pi/120)*i) sin((pi/120)*i) 0;
         -sin((pi/120)*i) cos((pi/120)*i) 0;
         0 0 1];
    metasx = affine2d(a);
    temp = imwarp(ball1,metasx);
    spin{i+1} = temp;
end

%ginetai kai edw kanonikopoihsh tw n mege8wn tw n eikonwn pou
prokryptoun apo
%th peristrofh ths eikonas ths mpalas
for ii = 1:200
    [x,y,~] = size(spin{ii,1});
    if x-100>0
        cr = (x-100)/2;
        cr = floor(cr);
        spin{ii,1} = spin{ii,1}((cr+1):x-(cr+1),(cr+1):x-(cr+1),:);
    elseif x-100==0
        spin{ii,1} = spin{ii,1};
    end
end

%edw pleon ginetai o metasxhmatismos gia th kinhsh oloklhrhs ths
mpalas
%mesa sth paralia gia auto kai autes oi parametroi sth 3h grammh tou
pinaka
%a2 (sth matlab ekei ginetai to translation)
for i = 0:199
    i2 = 200;
    a2 = [1 0 0;
          0 1 0;
          40+3*i 3/2*ty-3/2*abs(i2-i)*abs(cos((i2-i)/16)) 1];
    metasx2 = affine2d(a2);
    temp2 = imwarp(spin{i+1},metasx2,'OutputView',rout);
    beach_vid{i+1} = temp2;
end

%dhmiourgia tw n frames
for i = 1:200
    beach_vid{i,1} = (1-beach_vid{i,1}(:,:,:)).*beach(:,:,:);
    beach_tel(i) = im2frame(uint8(round(255*beach_vid{i,1})),cmap);
end
implay(beach_tel,30)



---


%% script ekfylismos
% h mpala xanetai sto vathos

```



```

close all; clear all; clc

ball = rgb2gray(im2double(imread('ball.jpg')));
ball_mask = rgb2gray(im2double(imread('ball_mask.jpg')));
beach = rgb2gray(im2double(imread('beach.jpg')));

cmap = gray(256);

ball = imresize(ball,0.125);
ball_mask = imresize(ball_mask,0.125);
beach = imresize(beach,[600 1280]);
rout = imref2d(size(beach));
tx = mean(rout.XWorldLimits);
ty = mean(rout.YWorldLimits);

ball1 = (1-ball_mask).*(1-ball);

spin = cell([200,1]);
beach_vid = cell([200,1]);

%akolou8eitai h idia taktikh opws kai gia th paralia prwta h
peristrofh ths
%mpalas
for i = 0:199
    a = [cos((pi/120)*i) sin((pi/120)*i) 0;
        -sin((pi/120)*i) cos((pi/120)*i) 0;
        0 0 1];
    metasx = affine2d(a);
    temp = imwarp(ball1,metasx);
    %spin{i+1,1} = (uint8(round(255*temp)));
    spin{i+1} = temp;
end

%kanonikopoihsh mege8wn tw n eikonwn
for ii = 1:200
    [x,y,~] = size(spin{ii,1});
    if x-100>0
        cr = (x-100)/2;
        cr = floor(cr);
        spin{ii,1} = spin{ii,1}((cr+1):x-(cr+1),(cr+1):x-(cr+1),:);
    elseif x-100==0
        spin{ii,1} = spin{ii,1};
    end
end

%h epi8ymhth kinhsh ths mpalas sth paralia
for i = 0:199
    i2 = 200;
    % to b einai mia metablhth pou pros8esa gia na e3afanistei h
    mpala,
    % exei thn idiothta otan ginei ish me to 0 na parameinei mhden
    giati
    % alliws ginotan pali 1 eite eteine pros to 1 opote h mpala
    % e3afanizotan kai emfanizotan pali
    b = (1-(0.01*(i2-i/1.5)));
    if b<=0
        b = b;
    elseif b>0
        b = 0;
    end
end

```

```

    a2 = [b 0 0;
          0 b 0;
          abs(tx/4+2.8*i) 4.8/3*ty-((i+1)+1/1.7*abs((i2-
i)/1.7)*abs(cos(3*(i2-i)/16))) 1];

    metasx2 = affine2d(a2);
    temp2 = imwarp(spin{i+1},metasx2,'OutputView',rout);
    beach_vid{i+1} = temp2;
end

%dhmioyrgia twm frames
for i = 1:200
    beach_vid{i,1} = (1-beach_vid{i,1}(:,:,:)).*beach(:,:,:);
    beach_tel(i) = im2frame(uint8(round(255*beach_vid{i,1})),cmap);
end
imshow(beach_tel,15)

```