"In The Name of God"

Design by Mohammadreza

Arani

810196405

Associated TA: Ali Ranjbar& Hoda Barkhordarpoor

Chief TA: D.r jamal kazazi

Subject:Shazam App

Sources: My own hardwork&creativity , Internet sites such as Mathworks and good friends

Chapters:

First_part:

As we reached phase 2 we considered making a database ,actually a little one, to get closer to shazam app in reality.

Here we go again having my codes and their description as always:

Let's start with hash_producer which is the main function used in making database:

As we can see down here, this function calls some other functions which are (peak_to_pair;matched;voiceprint) and some have been written for now and some are old ones. Peak_to_pair is given to us.Voice print has came from phase1.

Matched function:

```
function rightous_pairs=matched(pairs)

rightous_pairs(:,1)=pairs(:,3);
rightous_pairs(:,2)=pairs(:,4);
rightous_pairs(:,3)=pairs(:,1);
rightous_pairs(:,4)=pairs(:,2)-pairs(:,1);
end
```

```
function  hash_producer(audio,name)
% song_list = get_mp3_list('C:\Users\Mohammad Reza\Desktop\CAsignalp2\Songs\Train'
% songname=song_list(1,1);
% audio=songname{1};
% name=songname{1};
[data,fs1]=audioread(audio);

[peaks,T,F]=voiceprint(data,fs1);

pairs = peak_to_pair(peaks);

rightous_pairs=matched(pairs);

L1=length(rightous_pairs(:,1));

hash1=zeros(L1,2);
for i = 1 : L1
     hash1( i , 1 )=hash_func(rightous_pairs(i,1),rightous_pairs(i,2),rightous_pai
     hash1( i  , 2)=(rightous_pairs(i,3));
end

hashed=unique(hash1(:,1));
L2=length(hashed(:,1));
hash2=zeros(L2,L2);

for j= 1 : L2
    n=1;
    hash2(j,1) = hashed(j,1);
    for c= 1 : L1
        if(hash2( j,1)== hash1(c,1))
            n=n+1;
         hash2(j,n)= hash1(c,2);
        end
    end
end

name2=strcat(name,'_HASHTABLE.mat');
save( name2 ,'hash2');

end
```

We read the audio and then finding peaks and making them as ordered in the description with name righteous_pairs then beginning to making hash_table using hash_func in the chapter 2 and then memorizing the ts1 for each peak,

Saving them by given name to the function with some changes into it to be compatible with the big big picture out of project.

We use this progress in a for loop to have this table for all parts of our Songs>>Train step.

The output is saved anywhere wanted to be used

Later.

```
clc;
clear;

song_list = get_mp3_list('C:\Users\Mohammad Reza\Desktop\CAsignalp2\Songs\Train');
for i=1:length(song_list(:,1))
songname=song_list(i,1);
hash_producer(songname{1},num2str(i));
end
```

Chapter2 Hash_func

We just wrote the code you suggested to do then we named it hash_func which comes down here:

```
function index=hash_func(f1,f2,t_var)
index=(t_var)*2^16+f1*2^8+f2;
end
```

It's all needed for this part and nothing more.

Chapter3 Matching Clips

To find matches we needed to find and compare datas so we needed a search algorithm to find those and we could use much more better functions but we prefered to use 2 consecutive for loops which is the worst but the most simple one.

Searcher is the name we put on it:

It gives out chance which is the number of

Paired data in comparison with given file.

```
function  chance = searcher(Clip,hashing,SNR)
    [data,fs] = audioread(Clip);
    data=awgn(data , SNR );
    peaks = voiceprint(data,fs);
    pairs = peak_to_pair(peaks);
    rightous_pairs=matched(pairs);
    L1 = length(rightous_pairs);
    chance = 0;


    for i = 1 : L1
        hash1 = hash_func(rightous_pairs(i,1),rightous_pairs(i,2),rightous_pairs(i,
        p=0;
       for p=1:length(hashing)
            if(hashing(p,1)==hash1)
               chance = chance+1;
            end
        end
     end
```

Next step was just to get the clip and check it with our tables so we wrote the function to do so;

First this function had another view then we made it better and compatible with our thoughts in other steps.

Chapter4 Running and getting the output

Here we have the last script needed for running and having the output.

```
clc;
clear;
song_list = get_mp3_list('C:\Users\Mohammad Reza\Desktop\CAsignalp2\Songs\Clips');
L=length(song_list(:,1));

for j=1:L
    Clip = song_list{j};


% Clip = 'Khob_Shod-4.mp3';

SNR=-15:3:15;


for i=1:size(SNR,2)
  disp(i);
match_clip(Clip,SNR(1,i));

end

end
```

Match_clip function:

```
function match_clip(Clip,SNR)



%  Clip = 'Norouz-5.mp3';

song_list = get_mp3_list('C:\Users\Mohammad Reza\Desktop\CAsignalp2\Songs\Train');
%%save('SONGID_DB.mat','song_list');
%%SONGID_DB=load('SONGID_DB.mat');

% find(strcmp(SONGID_DB2(:,1),'khobshod.mp3'));
% find(strcmp(SONGID_DB2(:,1),'Norouz.mp3'));
% find(strcmp(SONGID_DB2(:,1),'Iraneman.mp3'));
L=length(song_list(:,1));
chance=zeros(1,L);

for i=1:L
    hashed=load(strcat(num2str(i),'_HASHTABLE.mat'));
    hashing=hashed.hash2;
chance(1,i) = searcher(Clip,hashing,SNR);
end
% hashed=load('hash_line1.mat');
% hashing=hashed.hash2;
% chance1 = searcher(Clip,hashing,SNR);
%
%
% hashed=load('hash_line2.mat');
% hashing=hashed.hash2;
% chance2 = searcher(Clip,hashing,SNR);
%
%
%
% hashed=load('hash_line3.mat');
% hashing=hashed.hash2;
% chance3 = searcher(Clip,hashing,SNR);

% songname='';

% if(chance1 > chance2)&&(chance1 > chance3)
%
%     songname=song_list(1,1);
% elseif(chance2 > chance1)&&(chance2 > chance3)
%
%     songname=song_list(2,1);
%
% elseif(chance3 > chance1)&&(chance3 > chance2)
%
%     songname=song_list(3,1);
%
% end

for j=1:L
    if(chance(1,j)==max(chance))
        disp('The input song you were looking for is: ');
        disp(song_list(j,1));
    end
end

% disp('The input song you were looking for is: \n');
% disp(songname);
 end
```

The result is out here with the noise added from -15 to 15 added 3 by 3 for each song:

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
1	Clips	Irane_Mal	Irane_Mal	Irane_Mal	Irane_Man	-5-32	Khob_Sho	Khob_Sho	Khob_Sho	Khob_Shod	-4	Norouz-1-	Norouz-2	Norouz-3-	Norouz-5						
2	SNR																				
3	-15	T	T	T	T		F	F	F	F		T	T	T	F						
4	-12	T	T	T	T		F	F	F	F		T	T	T	F						
5	-9	T	T	T	T		T	F	T	F		T	T	T	T						
6	-6	T	T	T	T		T	T	T	T		T	T	T	T						
7	-3	T	T	T	T		T	T	T	T		T	T	T	T						
8	0	T	T	T	T		T	T	T	T		T	T	T	T						
9	3	T	T	T	T		T	T	T	T		T	T	T	T						
10	6	T	T	T	T		T	T	T	T		T	T	T	T						
11	9	T	T	T	T		T	T	T	T		T	T	T	T						
12	12	T	T	T	T		T	T	T	T		T	T	T	T						
13	15	T	T	T	T		T	T	T	T		T	T	T	T						

END Of The Semester

“Happy finishing Term4”