Re: Halit et al. (2004) - question about randomizing phase spectra in an image

Gergely Csibra <CsibraG@ceu.edu>

Thu 06/10/2022 15:05

To: Ganea, Natasa <natasa.ganea@yale.edu>

Dear Natasa,

It's good to hear from you!

I hardly remember what we did 20 years ago, but on the basis of the footnote and your script I reconstructed what we must have done (see below).

It is possible that we performed the transform separately for the three colour components and then reunited them, but in principle the present version should do the same (if I understand correctly the description of the fft2 function: "If X is a multidimensional array, then fft2 takes the 2-D transform of each dimension higher than 2.") But you can verify this.

I would advise you to use images with white background or your transformed image may turn out to be too dark.
Also, by hindsight, perhaps you would get a more appropriate control image by shuffling the phase array rather than replacing it with random numbers.

All the best,
Gergo

***

```
%% clean up workspace
clc;
clearvars;

%% read & show image
i1=imread('bear.jpg');
figure(1),imshow(i1);

%% do 2-dim Fourier Transform of the image & extract magnitude and phase
f1=fft2(i1);
mag1=real(abs(f1)); % extract magnitude %% I took the real component only
phase1 = real(angle(f1)); % extract phase %% I took the real component only

%% change the phase
%phase1=exp(normrnd(0,0.5).*phase1); % multiply phase with a random number -0.5 & 0.5, calculate exponential

%instead of the above, this is what we specified in the footnote
phase1=(rand(size(phase1))-0.5)*pi; %if you comment out this line, you should get back the original image
```

```
%% do inverse Fourier Transform of both magnitude and phase
%a1= fft2(ifft2(mag1));
%a2= fft2(ifft2(phase1));

%% reconstruct the image, then apply inverse Fourier Transform
%a3= a1.*a2;

ar = mag1.*cos(phase1); %calculate the real component with the new phase
ai = mag1.*sin(phase1); %calculate the imag compnent og the new phase
a3 = complex(ar,ai); %reconstruct the Fourier transform

a4=real(ifft2(a3)); %run the inverse transform
figure(2),imshow(uint8(a4))
```

On 6 Oct 2022, at 19:10, Ganea, Natasa <natasa.ganea@yale.edu> wrote:

Dear Prof. Csibra,

I hope my email finds you well. I am writing because I am trying to replicate the randomised phase spectra image manipulation that you employed in some of your papers (Halit et al., 2004, Gliga et al., 2009).

In Halit et al., you detail in Footnote 1, how the randomisation of the phase spectra was done. I tried to implement that in my Matlab script, but it does not work, and I do not know what I am doing wrong. The closest I got to your image manipulation was when I multiplied the phase with a random number between -0.5 and 0.5.

I copied below the footnote, and my Matlab script. I would be very grateful if you could give me some feedback.

Thank you very much, and I hope you do not mind me reaching out regarding this image manipulation.

Best wishes,
Natasa

--
***Natasa Ganea**, PhD*
she/her/hers
Postdoctoral Research Fellow
Haskins Laboratories

Footnote:
"Specifically, (1) a two-dimensional Fast Fourier Transformation was applied to all three colour components of the images, (2) the phase on each frequency was replaced by a

random value between −π and π (uniform distribution), (3) an inverse Fourier transformation reconstituted the image, (4) to which the outer contour of the original face was applied as a mask."

Matlab Script:

```
%% clean up workspace
clc;
clearvars;

%% read & show image
i1=imread('bear.jpg');
figure(1),imshow(i1);

%% do 2-dim Fourier Transform of the image & extract magnitude and phase
f1=fft2(i1);
mag1=abs(f1); % extract magnitude
phase1 = angle(f1); % extract phase

%% change the phase
phase1=exp(normrnd(0,0.5).*phase1); % multiply phase with a random number -0.5 & 0.5, calculate exponential

%% do inverse Fourier Transform of both magnitude and phase
a1= fft2(ifft2(mag1));
a2= fft2(ifft2(phase1));

%% reconstruct the image, then apply inverse Fourier Transform
a3= a1.*a2;
a4=ifft2(a3);
figure(2),imshow(uint8(a4))
```

<bear.jpg>