Wykład wstępny (II) *<< zrealizowany w 2006, przedrostki-nazwy zretuszowane w Acrobat >>*

z Podstaw Przetwarzania Informacji

(na danych obrazów 2D w środowisku Matlab 6.x 7.x)

Tytułem przypomnienia listy toolbox'ów zainstalowanych (lub tylko dostarczanych) razem z pakietem Matlab (w wersji 6.5), podano wydruk:

```
Ver
```

```
---------------------------------------------------------------------------------------
MATLAB Version 6.5.0.180913a (R13)
MATLAB License Number: 0
Operating System: Microsoft Windows XP Version 5.1 (Build 2600: Dodatek Service Pack 2)
Java VM Version: Java 1.3.1_01 with Sun Microsystems Inc. Java HotSpot(TM) Client VM
---------------------------------------------------------------------------------------
MATLAB                                Version 6.5       (R13)
Simulink                              Version 5.0       (R13)
Aerospace Blockset                    Version 1.0.1     (R13)
CDMA Reference Blockset               Version 1.1       (R13)
Communications Blockset               Version 2.5       (R13)
Communications Toolbox                Version 2.1       (R13)
Control System Toolbox                Version 5.2       (R13)
Curve Fitting Toolbox                 Version 1.1       (R13)
DSP Blockset                          Version 5.0       (R13)
Data Acquisition Toolbox              Version 2.2       (R13)
Database Toolbox                      Version 2.2.1     (R13)
Datafeed Toolbox                      Version 1.3.1     (R13)
Dials & Gauges Blockset               Version 1.1.2     (R13)
Embedded Target for Motorola MPC555   Version 1.0.1     (R13)
Embedded Target for Texas Instrumen...  Version 1.0     (R13)
Excel Link                            Version 2.0       (R13)
Filter Design Toolbox                 Version 2.2       (R13)
Financial Derivatives Toolbox         Version 2.0       (R13)
Financial Time Series Toolbox         Version 2.0       (R13)
Financial Toolbox                     Version 2.2.1     (R13)
Fixed-Point Blockset                  Version 4.0       (R13)
Fuzzy Logic Toolbox                   Version 2.1.2     (R13)
GARCH Toolbox                         Version 1.0.2     (R13)
Image Processing Toolbox              Version 3.2       (R13)
Instrument Control Toolbox            Version 1.2       (R13)
LMI Control Toolbox                   Version 1.0.8     (R13)
MATLAB COM Builder                    Version 1.0       (R13)
MATLAB Compiler                       Version 3.0       (R13)
MATLAB Excel Builder                  Version 1.1       (R13)
MATLAB Link for Code Composer Studi...  Version 1.0     (R13)
MATLAB Report Generator               Version 1.3       (R13)
MATLAB Runtime Server Development Kit  Version 6.1.1    (R13)
MATLAB Web Server                     Version 1.2.2     (R13)
Mapping Toolbox                       Version 1.3       (R13)
Model Predictive Control Toolbox      Version 1.0.7     (R13)
Model-Based Calibration Toolbox       Version 1.1       (R13)
Mu-Analysis and Synthesis Toolbox     Version 3.0.7     (R13)
Neural Network Toolbox                Version 4.0.2     (R13)
Nonlinear Control Design Blockset     Version 1.1.6     (R13)
Optimization Toolbox                  Version 2.2       (R13)
Partial Differential Equation Toolbox  Version 1.0.4    (R13)
Real-Time Windows Target              Version 2.2       (R13)
Real-Time Workshop                    Version 5.0       (R13)
Real-Time Workshop Embedded Coder     Version 3.0       (R13)
Requirements Management Interface     Version 1.0.4     (R13)
Robust Control Toolbox                Version 2.0.9     (R13)
SB2SL (converts SystemBuild to Simu...  Version 2.5     (R13)
Signal Processing Toolbox             Version 6.0       (R13)
SimMechanics                          Version 1.1       (R13)
SimPowerSystems                       Version 2.3       (R13)
Simulink Performance Tools            Version 1.2       (R13)
Simulink Report Generator             Version 1.3       (R13)
Spline Toolbox                        Version 3.1.1     (R13)
Stateflow                             Version 5.0       (R13)
Stateflow Coder                       Version 5.0       (R13)
Statistics Toolbox                    Version 4.0       (R13)
Symbolic Math Toolbox                 Version 2.1.3     (R13)
System Identification Toolbox         Version 5.0.2     (R13)
Virtual Reality Toolbox               Version 3.0       (R13)
Wavelet Toolbox                       Version 2.2       (R13)
xPC Target                            Version 2.0       (R13)
xPC Target Embedded Option            Version 2.0       (R13)
```

W dalszej części omawiane będą zasadniczo toolbox'y:

```
Image Processing Toolbox
Signal Processing Toolbox
Wavelet Toolbox
```

Pakiet Matlab'a dysponuje szerokim zbiorem danych reprezentatywnych, wykorzystywanych w demonstrowaniu możliwości obliczeniowych przetwarzania danych. Są to dla przykładu dane obrazów płaskich 2D intensywności luminancji oraz grafiki kolorowej:

```
>> load woman
```

(z domyślnym określeniem gdzieś tam zasobu `woman2.mat`)lub dokładniej:

```
>> load 'c:\MATLAB6p5\toolbox\wavelet\wavedemo\woman.mat'
```

lub/oraz dogodniej, dla późniejszej obróbki danych, jest własnoręczne przytoczenie pliku z rozszerzeniem z zasobów pomocy środowiska Matlab:

```
>>obraz1=imread('c:\MATLAB6p5\help\toolbox\wavelet\woman.gif');
```

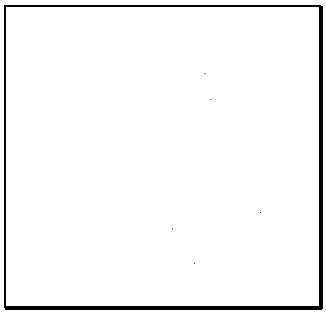
Drugie polecenie, choć wymaga szczegółowej znajomości rozszerzenia pliku oraz jego lokalizacji, daje podstawy do jednoznacznego odwoływania się do jego zawartości z przestrzeni roboczej środowiska Matlab:

```
>> whos
```

```
  Name        Size                Bytes  Class

  X         128x128              131072  double array
  map       255x3                  6120  double array
  obraz1    324x471              152604  uint8 array

Grand total is 169753 elements using 289796 bytes

>>
```

Powyżej widać na wydruku rezultatów (w ramce), zmienną `X` typu `double float precision`, co w wyświetleniu *bezpośrednim* danych daje mierny wynik:

```
>> imshow(X)
```



Rys.1 Wyświetlanie z użyciem polecenia `imshow` bezpośrednio zmiennych typu `double` zawodzi!

```
>> imshow(uint8(X));
```

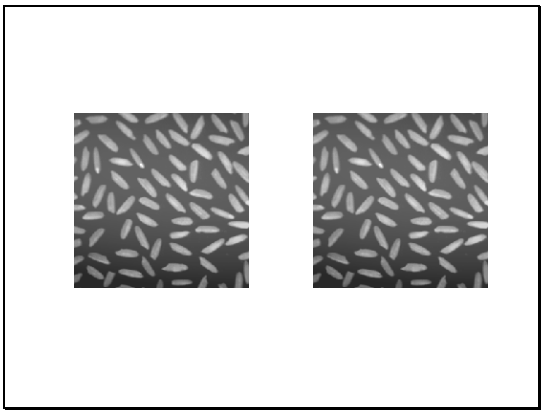
Rys.2 Wyświetlanie z użyciem polecenia `imshow` z rzutowaniem do typu `uint8` z `double` skutkuje!

Inny przykład to obraz zbioru ziaren ryżu, o danych wczytywanych odpowiednio z domyślną i szczegółową ścieżką dostępu:

```
>> im2=imread('rice.tif');
>> IM2=imread('c:\MATLAB6p5\toolbox\images\imdemos\rice.tif');
```

Zaznaczyć należy, że obecnie pod dwiema różnymi zmiennymi IM2 oraz im2 znajdują się dane tego samego obrazu rice.tif:

```
>> figure,subplot(1,2,1);imshow(im2);subplot(1,2,2);imshow(IM2);
```



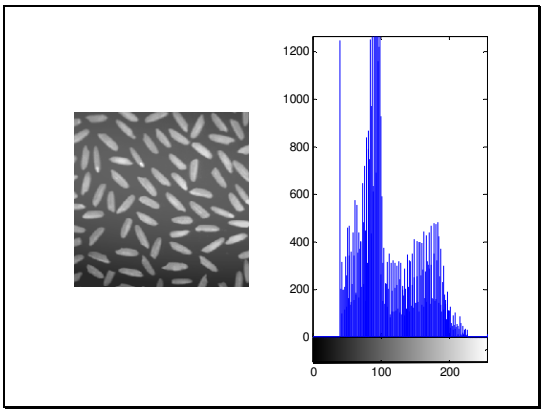
Rys.3 Wyświetlanie z użyciem polecenia `imshow` *oraz polecenia* `subplot` *prezentacji w podoknach!*

Co *widać* na powyższym rysunku numer 3?

Ano, po pierwsze zachowana jest dość duża wyrazistość, tj. kontrastowość białych obiektów (ziaren ryżu) na znacznie ciemniejszym tle. Jednak, tło przy tym jest o nierównomiernej intensywności luminancji, co wskazuje na częściowe niedoświetlenie scenerii.

Wyświetlenie histogramu dla tego zbioru danych, to:

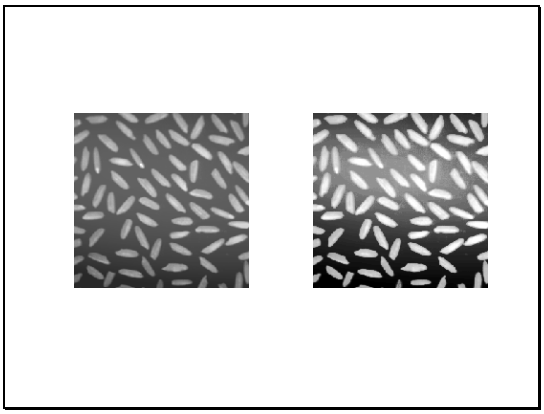
```
>> figure,subplot(1,2,1);imshow(im2);subplot(1,2,2);imhist(im2);
```



Rys.4 Wyświetlanie jednoczesne treści obrazu i jego histogramu.

Czasami, operacja na danych obrazów 2D z analizą histogramową dają poprawienie *czytelności* lub *przejrzystości* obrazu:

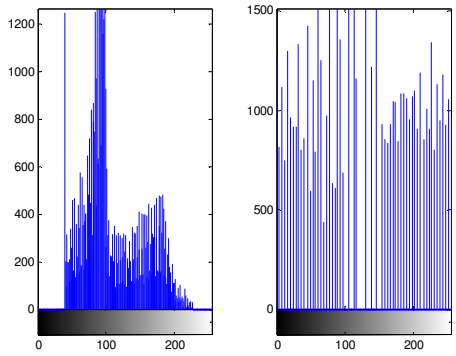
```
>> im2eq=histeq(im2);
>> figure,subplot(1,2,1);imshow(im2);subplot(1,2,2);imshow(im2eq);
```



Rys.5 Po lewej obraz pierwotny `rice.tif`, po prawej z wyrównanym przebiegiem histogramu

Jak widać funkcja `histeq` może być z powodzeniem wykorzystywana w *uwypuklaniu* szczegółów, niuansów, detali na obrazie, takich jak lokalne *niedoświetlenia* scenerii w polu ujęcia. Jeszcze tylko, poniżej, zamieszczone zostaną przebiegi histogramów:

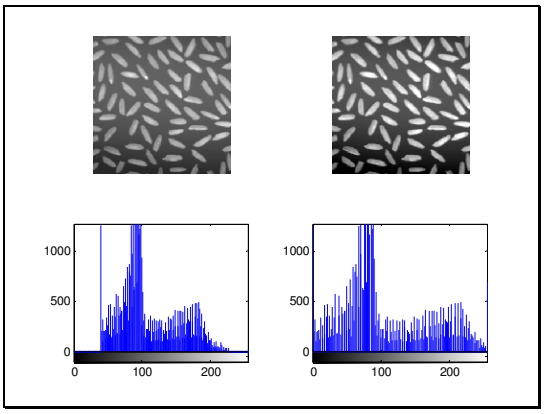
```
>> figure,subplot(1,2,1);imhist(im2);subplot(1,2,2);imhist(im2eq);
```



Rys.5 Po lewej histogram obrazu pierwotnego `rice.tif`, po prawej histogram wyrównany

Zrównoważenie histogramu (choć bardziej *subtelne* w działaniu), nie realizuje tego co może zapewnić wywołanie funkcji `imadj` w połączeniu z detekcją zakresu zmian intensywności luminancji w obrazie, tj. z użyciem funkcji `stretchlim`:

```
figure,subplot(2,2,1);imshow(im2);subplot(2,2,2);imshow(im2adj);...
subplot(2,2,3);imhist(im2);subplot(2,2,4);imhist(im2adj);
```



Rys.6 Górny wiersz:obraz pierwotny `rice.tif`, oraz po skontrastowaniu
dolny wiersz: histogram obrazu pierwotnego, oraz po skontrastowaniu.

W przypadku grafiki kolorowej w reprezentacji *RGB*, mamy do czynienia ze swoistego rodzaju trójwymiarową tablicą! Zabawa z poszerzaniem kontrastu dla intensywności luminancji niezależnie w każdym z trzech kanałów barw może się skończyć następująco:

```
>> RGB1 = imread('flowers.tif');
>> RGB2 = imadjust(RGB1,[.1 .1 .1; .2 .2 1],[]);
```

przy działaniach na dwóch podoknach jednego wykresu:

```
>> figure,subplot(1,2,1);imshow(RGB1);subplot(1,2,2);imshow(RGB2);
```

Rys.7 Po lewej: obraz oryginalny `flowers.tif`
po prawej: obraz po przeskalowaniu intensywności luminancji w trzech kanałach.

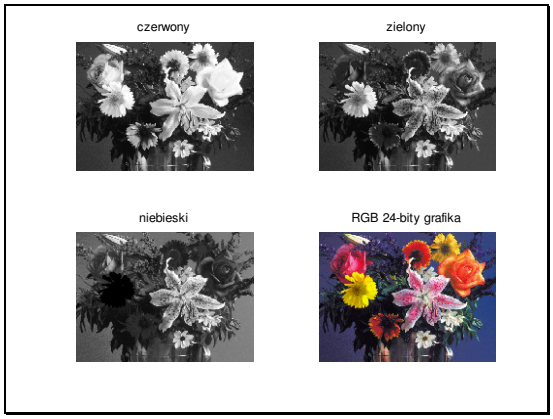
```
>> whos RGB*
```

Name	Size	Bytes	Class
RGB1	362x500x3	543000	uint8 array
RGB2	362x500x3	543000	uint8 array

Grand total is 1086000 elements using 1086000 bytes

Oprócz powyższych uwag należy jeszcze przytoczyć możliwość, *drastycznego* okrajania informacji służącej w manipulacjach na danych intensywności obrazów szarości lub kolorowych w celu, odcisku *negatywowego* lub zwyczajnej *pastelizacji* (czytaj *stylizacji)* treści obrazu. *Apropos*, reprezentacja danych koloru to tylko rozwinięcie formatu danych intensywności luminancji w formie trójwymiarowej tablicy liczb `uint8`:

```
>> RGB_R=RGB1(:,:,1); %kanał czerwony
>> RGB_G=RGB1(:,:,2); %kanał zielony
>> RGB_B=RGB1(:,:,3); %kanał niebieski
```



Rys.7B Składowe RGB obrazu `flowers.tif`

Obecnie omówione zostaną pewne zastosowania funkcji skalowania rozmiarem obrazu płaskiego 2D, z pomocą funkcji `imresize`. Istnieje możliwość dowolnego przeskalowywania rozmiarów pierwotnych obrazów płaskich 2D z naruszaniem ich proporcji wymiarów, w operacjach powiększania i pomniejszania detali, z mniejszą i większą w założeniach dokładnością ich skalowania.

```
>> im2x2=imresize(im2,[512 512],'bicubic');
>> IM2X2=imresize(IM2,2,'bicubic');
```

Ano, na powyższym przykładzie dwóch poleceń, zaprezentowano dwie składnie polecenia dwukrotnego zwiększenia rozmiaru obrazu pierwotnego, przy użyciu dokładniejszej metody skalowania `bicubic` (metody interpolacji dwusześciennej):

```
>> whos im* IM*
  Name        Size                Bytes  Class

  IM2       256x256             65536  uint8 array
  IM2X2     512x512            262144  uint8 array
  im2       256x256             65536  uint8 array
  im2x2     512x512            262144  uint8 array
```

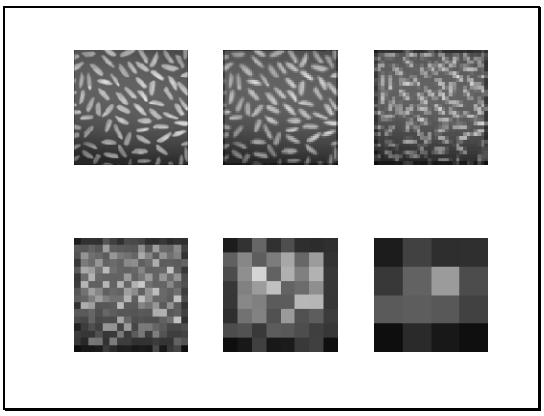
Jak widać powyżej `IM2` oraz `im2` przeskalowane zostały w transformacji do obrazów 512x512 pikseli, `IM2x2` oraz `im2x2`, odpowiednio, o identycznej treści.
Gdyby jednak skalowanie realizować w stronę niższej reprezentacji pikselowej:

```
>> im2div02=imresize(im2,0.5,'bicubic');
>> im2div04=imresize(im2,0.25,'bicubic');
>> im2div08=imresize(im2,0.125,'bicubic');
>> im2div16=imresize(im2,0.0625,'bicubic');
>> im2div32=imresize(im2,0.03125,'bicubic');
>> im2div64=imresize(im2,0.015625,'bicubic');
>> whos im2div*
```

```
  Name          Size                Bytes  Class
  im2div02    128x128             16384  uint8 array
  im2div04    64x64                4096  uint8 array
  im2div08    32x32                1024  uint8 array
  im2div16    16x16                 256  uint8 array
  im2div32    8x8                    64  uint8 array
  im2div64    4x4                    16  uint8 array
Grand total is 21840 elements using 21840 bytes
```

,to zasadniczo otrzymuje lokalne *uśrednienie* luminancji na obrazie:

```
>> figure,subplot(2,3,1);imshow(im2div02);subplot(2,3,2);…
imshow(im2div04);subplot(2,3,3);imshow(im2div08);...
subplot(2,3,4);imshow(im2div16);subplot(2,3,5);…
imshow(im2div32);subplot(2,3,6);imshow(im2div64);
```



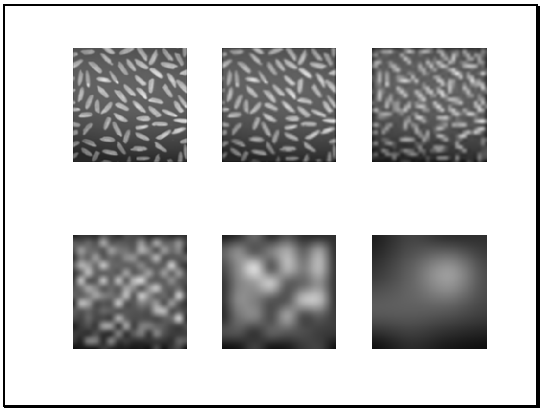
Rys8 Kolejno od lewej u góry oryginalny obraz `rice.tif`
sukcesywnie 2-krotnie zmniejszany w jego reprezentacji pikselowej.

Najciekawszy jednak efekt w całości zabiegów, jak i najbardziej praktyczny otrzymuje się przy ponownym przeskalowywaniu wszystkich bitmap do rozmiaru oryginalnego 256x256 pikseli:

```
>> im2copy02=imresize(im2div02,2,'bicubic');
>> im2copy04=imresize(im2div04,4,'bicubic');
>> im2copy08=imresize(im2div08,8,'bicubic');
>> im2copy16=imresize(im2div16,16,'bicubic');
>> im2copy32=imresize(im2div32,32,'bicubic');
>> im2copy64=imresize(im2div64,64,'bicubic');
>> whos im2copy*
```

Name	Size	Bytes	Class
im2copy02	256x256	65536	uint8 array
im2copy04	256x256	65536	uint8 array
im2copy08	256x256	65536	uint8 array
im2copy16	256x256	65536	uint8 array
im2copy32	256x256	65536	uint8 array
im2copy64	256x256	65536	uint8 array

Grand total is 393216 elements using 393216 bytes



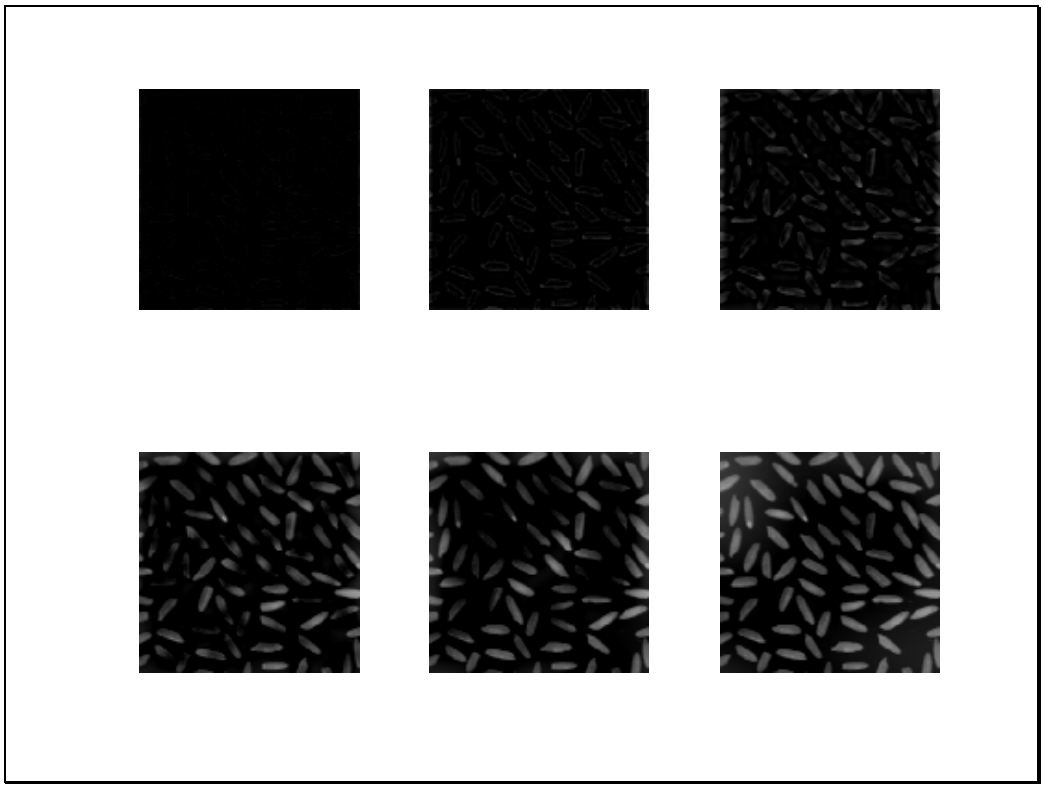
Rys9 Kolejno od lewej u góry kopie obrazu `rice.tif` sukcesywnie powiększane do rozmiarów oryginalnych.

Na rysunku 9 widać, jak ponowna zmiana rozmiarów kopii pomniejszonych obrazu `rice.tif`, zlikwidowała efekt pikselizacji. Natomiast rozmyta postać zmian intensywności luminancji może obecnie posłużyć do niwelacji nierównomierności oświetlenia:

```
>> im2_sub02=imsubtract(im2,im2copy02);
>> im2_sub04=imsubtract(im2,im2copy04);
>> im2_sub08=imsubtract(im2,im2copy08);
>> im2_sub16=imsubtract(im2,im2copy16);
>> im2_sub32=imsubtract(im2,im2copy32);
>> im2_sub64=imsubtract(im2,im2copy64);
```

Nadto, jeszcze potrzebna jest pewna prezentacja na 6 podwykresach wyników:

```
>> figure,subplot(2,3,1);imshow(im2_sub02);…
subplot(2,3,2);imshow(im2_sub04);subplot(2,3,3);imshow(im2_sub08);...
subplot(2,3,4);imshow(im2_sub16);subplot(2,3,5);…
imshow(im2_sub32);subplot(2,3,6);imshow(im2_sub64);
```



Rys9 Kolejno od lewej u góry kopie obrazu `rice.tif`
z sukcesywnie ujmowaną szacowaną nierównomiernością tła!.

Wracając do kopii obrazu `rice.tif` o oznaczeniach `im2copy*`, można obecnie '*zgotować*' garść ryżu, w formie animacji, czy to pod nazwą zmiennej `M` w przestrzeni roboczej Matlab'a, czy to również w formacie animacji `avi` na dysku w katalogu roboczym:

```
>> M=moviein(7);
>> imshow(im2);
>> M(:,1)=getframe;
>> imshow(im2copy02);
>> M(:,2)=getframe;
>> imshow(im2copy04);
>> M(:,3)=getframe;
>> imshow(im2copy08);
>> M(:,4)=getframe;
>> imshow(im2copy16);
>> M(:,5)=getframe;
>> imshow(im2copy32);
>> M(:,6)=getframe;
>> imshow(im2copy64);
>> M(:,7)=getframe;
>> whos M
```

```
  Name      Size                Bytes  Class

  M         1x7               1387605  struct array

Grand total is 1387043 elements using 1387605 bytes
```

Teraz oto, ryż '*gotuje się*' na naszych oczach:

```
>> movie(M);
```

, a obecnie również poddawany jest *cieplnej obróbce* na *'wrzątku'* w formie animacji `avi`:

```
>> movie2avi(M,'Gotuj_Ryz_w_Zlepke.avi');
```

```
Warning: The frame height has been padded to be a multiple of four as required by Intel Indeo.
> In C:\MATLAB6p5\toolbox\matlab\iofun\@avifile\addframe.m at line 139
  In C:\MATLAB6p5\toolbox\matlab\iofun\movie2avi.m at line 67
Warning: The frame width has been padded to be a multiple of four as required by Intel Indeo.
> In C:\MATLAB6p5\toolbox\matlab\iofun\@avifile\addframe.m at line 145
  In C:\MATLAB6p5\toolbox\matlab\iofun\movie2avi.m at line 67
```

Jednakże, brakuje tworzenia animacji w prezentacji wyników końcowych w pętli `for`:

```
MWyn=moviein(7);
MacierzWyn=[im2_sub02;im2_sub04;im2_sub08;im2_sub08;…
im2_sub16;im2_sub32;im2_sub64;im2];
for i=1:7,
    imshow(MacierzWyn((1+(i-1)*256):(i*256),1:256));
    MWyn(:,i)=getframe;
end;
movie(Mwyn);
```

Oto jeszcze tylko zapis na zwolnionej klatce (*2 klatki na sek.*) na dysk całości wyników:

```
>> movie2avi(MWyn,'Odcedzaj_Ryz_ze_Wrzatku.avi','COMPRESSION','Indeo3','FPS',2);
```

```
Warning: The frame height has been padded to be a multiple of four as required by Intel Indeo.
> In C:\MATLAB6p5\toolbox\matlab\iofun\@avifile\addframe.m at line 139
  In C:\MATLAB6p5\toolbox\matlab\iofun\movie2avi.m at line 67
Warning: The frame width has been padded to be a multiple of four as required by Intel Indeo.
> In C:\MATLAB6p5\toolbox\matlab\iofun\@avifile\addframe.m at line 145
  In C:\MATLAB6p5\toolbox\matlab\iofun\movie2avi.m at line 67
```

Suplement (do dalszej eksploracji *Image Processing Toolbox*):

```
>> lookfor image
```

```
CONTRAST Gray scale color map to enhance image contrast.
FRAME2IM Convert movie frame to indexed image.
IM2FRAME Convert indexed image into movie format.
IM2JAVA Convert image to Java image.
IMAGE  Display image.
IMAGESC Scale data and display as image.
IND2RGB Convert indexed image to RGB image.
PRINT Print figure or model. Save to disk as image or M-
file.
IMAGEVIEW Show image in figure window
HDFDF24 MATLAB gateway to HDF 24-bit raster image
interface.
HDFDFR8 MATLAB gateway to HDF 8-bit raster image
interface.
IMREAD Read image from graphics file.
IMWRITE Write image to graphics file.
imagedemo.m: %% Images and Matrices
 IMAGEEXT   Examples of images with a variety of colormaps
XPGALLRY Gallery image shell.
 AERO_POINTER_TRACKER demo script for generating an
optical image
SF This M-function is called in the absence of a valid
Stateflow image.
sf_cruise_button_mgr.m: % Load required images if they
don't exist
 RTW_C Creates the makefile used to build the RTW C code
image.
BLKPROC Implement distinct block processing for image.
BWAREA Compute the area of objects in binary image.
BWEULER Compute the Euler number of binary image.
BWFILL Fill background regions in binary image.
BWLABEL Label connected components in binary image.
BWLABELN Label connected components in N-D binary image.
BWMORPH Perform morphological operations on binary image.
BWPACK Pack binary image.
BWPERIM Find perimeter pixels in binary image.
BWSELECT Select objects in binary image.
BWUNPACK Unpack binary image.
CHECKERBOARD Create checkerboard image.
```

```
CMUNIQUE Find unique colormap colors and corresponding
image.
DECONVBLIND Image restoration using blind deconvolution
algorithm.
DECONVLUCY Image restoration using Lucy-Richardson
algorithm.
DECONVREG Image restoration using regularized filter.
DECONVWNR Image restoration using Wiener filter.
DICOMREAD  Read a DICOM image.
DICOMWRITE   Write images as DICOM files.
DILATE Perform dilation on binary image.
DITHER Convert image using dithering.
EDGE Find edges in intensity image.
EDGETAPER Taper the discontinuities along the image edges.
ERODE Perform erosion on binary image.
GETIMAGE Get image data from axes.
GRAY2IND Convert intensity image to indexed image.
GRAYSLICE Create indexed image from intensity image by
thresholding.
GRAYTHRESH Compute global image threshold using Otsu's
method.
IM2BW Convert image to binary image by thresholding.
IM2COL Rearrange image blocks into columns.
IM2DOUBLE Convert image to double precision.
IM2MIS Convert image to Java MemoryImageSource.
IM2UINT16 Convert image to sixteen-bit unsigned integers.
IM2UINT8 Convert image to eight-bit unsigned integers.
IMABSDIFF Compute absolute difference of two images.
IMADD Add two images, or add constant to image.
IMADJUST Adjust image intensity values or colormap.
IMAPPROX Approximate indexed image by one with fewer
colors.
IMCLEARBORDER Suppress light structures connected to image
border.
IMCLOSE Close image.
IMCOMPLEMENT Complement image.
IMCONTOUR Create contour plot of image data.
IMCROP Crop image.
IMDILATE Dilate image.
```

```
IMDIVIDE Divide two images, or divide image by constant.
IMERODE Erode image.
IMFEATURE Compute feature measurements for image regions.
IMFILL Fill image regions.
IMFILTER Multidimensional image filtering.
IMHIST Display histogram of image data.
IMLINCOMB Compute linear combination of images.
IMMOVIE Make movie from multiframe image.
IMMULTIPLY Multiply two images, or multiply image by
constant.  Z =
IMNOISE Add noise to image.
IMOPEN Open image.
IMRESIZE Resize image.
IMROTATE Rotate image.
IMSHOW Display image.
IMSLICE Return index into image deck.
IMSUBTRACT Subtract two images, or subtract constant from
image.
IMTRANSFORM Apply 2-D spatial transformation to image.
IND2GRAY Convert indexed image to intensity image.
IPTGETPREF Get Image Processing Toolbox preference.
IPTSETPREF Set Image Processing Toolbox preference.
ISBW Return true for binary image.
ISGRAY Return true for intensity image.
ISIND Return true for indexed image.
ISRGB Return true for RGB image.
LABEL2RGB converts label matrix to RGB image.
MAT2GRAY Convert matrix to intensity image.
MONTAGE Display multiple image frames as rectangular
montage.
PHANTOM Generate a head phantom image.
PIXVAL Display information about image pixels.
 README file for the Image Processing Toolbox.
REGIONPROPS Measure properties of image regions.
RGB2GRAY Convert RGB image or colormap to grayscale.
RGB2IND Convert RGB image to indexed image.
ROIFILL Smoothly interpolate within arbitrary image
region.
ROIFILT2 Filters a region of interest in an image.
STRETCHLIM Find limits to contrast stretch an image.
SUBIMAGE Display multiple images in single figure.
TIFFREAD Read a TIFF (Tagged Image File Format) file from
disk.
TIFFWRITE Write a TIFF (Tagged Image File Format) file to
disk.
TRUESIZE Adjust display size of image.
WARP Display image as texture-mapped surface.
WATERSHED Find image watershed regions.
DCTDEMO 2-D DCT image compression demo.
 IPEXSHEAR Image Padding and Shearing.
 IPEXTFORM Image Transform Gallery.
IMAGEM  Display a regular matrix map as an image
LOADCAPE load Cape Cod elevation image into a regular
matrix map
LOADMOONALB load moon albedo image into a regular matrix
map
 REPLACECOLOR  Replace a color in truecolor image with
another
xregCreateDisabledImage.m: %XREGCREATEDISABLEDIMAGE
 XREGIMAGE   Create a modified image object
DW2DIMGS Discrete wavelet 2-D image selection.
DW2DRWCD Discrete wavelet 2-D read-write Cdata for image.
IMGXTOOL Image extension tool.
WIMGCODE Image coding mode.
WPROPIMG Give image proportions.
DGUIIEXT Demonstrates Image extension GUI tools in the
Wavelet Toolbox.
```

```
imagecb.m: %GRAPH2d/IMAGECB   Callback function for
graph2d object
cfrimage.m: %   Image
GETIMGNAME returns a filename for use as a Report
Generator image
UPDATEPOINTER renews the pointer image of the component
CHECKFRAMEFORMAT returns true if the image format works
with printframes
CAPTURE Capture a Virtual Reality figure into a RGB image.
RESTOREUI Remove Images used to mimic user interface
controls in output.
hdfraster24info.m: %HDFRASTER8INFO Information about HDF
24-bit Raster image
HDFRASTER8INFO Information about HDF 8-bit Raster image
IMBMPINFO Get information about the image in a BMP file.
imcurinfo.m: %IMICURINFO Get information about the image
in a CUR file.
IMFTYPE Determine image file format.
IMHDFINFO Get information about images in an HDF file.
IMICOINFO Get information about the image in an ICO file.
IMPCXINFO Get information about the image in a PCX file.
IMPNMINFO Get information about the image in a PPM/PGM/PBM
file.
IMRASINFO Get information about the image in a RAS file.
IMXWDINFO Get information about the image in an XWD file.
READBMP Read image data from a BMP file.
READGIF Read an image from a GIF file.
READHDF Read an image from an HDF file.
READJPG Read image data from a JPEG file.
READPCX Read image data from a PCX file.
READPNG Read an image from a PNG file.
READPNM Read image data from a PPM/PGM/PBM file.
READRAS Read image data from a RAS file.
READTIF Read an image from a TIFF file.
READXWD Read image data from an XWD file.
RJPGC Read a JPEG image.
RTIFC Read a TIFF image.
WJPGC Write image data to a JPEG file.
WRITEHDF Write an raster-image HDF file to disk.
WTIFC Write image data to a TIFF file.
sf_root.m: % RESULT = SF_ROOT returns the root directory
where the Stateflow main image resides
IM2MIS Convert image to Java MemoryImageSource.
IM2UINT8 Convert image to eight-bit unsigned integers.
CHANGECLASS will change the storage class of an image.
DICOM_ENCODE_PIXEL_CELLS   Convert an image to pixel
cells.
DICOM_READ_IMAGE  Read the image data from a DICOM file.
dicom_set_image_encoding.m: %DICOM_SET_MMETA_ENCODING
Prepare for reading the image pixel data.
DICOM_XFORM_IMAGE   Reshape images, apply colorspace
transformations.
DITHERC Floyd-Steinberg image dithering (MEX function).
GET_IMAGE_TITLE gets title of image from its string name
or variable name
GRAYTO16 Scale and convert grayscale image to uint16.
GRAYTO8 Scale and convert grayscale image to uint8.
IMHISTC Image histogram.
IPTPREFS Image Processing Toolbox preference settings.
MORPHOP Dilate or erode image.
PR_BADIM  Load image data
PR_BADIM  Load image data
PR_BADIM  Load image data
PR_BADIM  Load image data
PR_BADIM  Load image data
PR_BADIM  Load image data
```

```
PR_BADIM  Load image data
```