Purpose: Learn how to perform data compression and decompression of grayscale image data.

Procedure: For *each* MATLAB m-file you write, be sure to comment the code sufficiently, and include initial comments as a crude help system. To level the playing field for storing the images, every team will store the images on disk to compare file size "before" and "after" compression in MATLAB's `*.MAT` format (using the `save` command with the "`-V6`" option). Type `help save` at the MATLAB command line for more details. The `-V6` option disables MATLAB from using it's own compression and Unicode encoding of the `*.MAT` file. **Note: You *must* use the `-V6` option, or your results will not be accepted.**

For this project, you are free to use (within reason) any programs you like in the Image Processing Toolbox, the Wavelet Toolbox, the Signal Processing Toolbox, or any other MATLAB Toolbox. To see what is available in these Toolboxes, start MATLAB and type `help images`, or `help wavelet`, or `help signal`, etc. You are also allowed to use code available on the course web page or on other web pages. Example MATLAB code on the course web page includes `norm2huff`, `huff2norm`, `norm2huff_16`, and `huff2norm_16`, which encode and decode Huffman coding on `uint8` and `uint16` symbols respectively.[1] This does *not* mean you can use, for example, a full JPEG compression program you found on the web! That would *not* be considered "within reason." If you use any code from other web pages, *you* are responsible for it's correct operation! Note that any code or background material that you use and that you didn't personally write *must* be cited properly.

You will be performing compression on a 512×512 8-bit (`uint8`) grayscale image (the ubiquitous Lena image). You choose which type or types of compression to implement, using any of the techniques (lossless or lossy) discussed in class or in Chapter 8 of your textbook, and any other image processing techniques we have discussed. You can choose to implement some or all of the traditional three steps of compression (see Figure 8.5 on p. 606 of your text), but whichever compression steps you do implement must be performed by either your own MATLAB code, or by a combination of your own and "borrowed" code. Be sure that for any "borrowed" code you use, that you briefly explain its operation in your report, and (if asked) are able to *fully explain* its operation to me in person.

You will also need to write the corresponding decompression MATLAB code. Upon decompression, your code should produce a visually acceptable 512×512 8-bit (`uint8`) grayscale image of Lena.

- ➪ You can find the image file already in `*.MAT -V6` format (as filename `Lena_Proj4_V6.mat`) on the course web page, in the Files⇒Images location. The `-V6` format must be used to prevent MATLAB from trying to add it's own type of compression, which would invalidate your results.
- ➪ Write two separate MATLAB programs, one to compress and one to decompress the Lena image. Save the compressed image data (including any needed overhead such as a codebook, dictionary, image dimensions, etc.) on disk in MATLAB's `*.MAT -V6` format.[2] Your compressed image data file needs to include in a single `*.MAT` file all the information needed for your decompression program to correctly produce a 512×512 8-bit (`uint8`) grayscale image of Lena. Before running your decompression program, be sure to execute the `clear` command in the MATLAB workspace to ensure your decompression program isn't making use of any lingering workspace variables. Your decompression

[1] Yes, the provided `norm2huff` and associated files only work on unsigned integers, so take that into consideration if you are using them to encode signed numbers such as DCT coefficients. A simple adjustment is necessary.

[2] You may want to type `help save` to see how to force the `-V6` format on a MATLAB MAT file.

program starts by loading the `*.MAT` file into an empty MATLAB workspace. All information that it needs to correctly decompress the image must be in that `*.MAT` file.

➪ In your project report results section, you should show two figures (as a minimum):

- the original Lena image, and
- the resulting Lena image after being compressed, saved as a `*.MAT -V6` file, loaded back into MATLAB, then decompressed.

The "before" and "after" images should be together, alone on a page, stacked vertically, with the "before" image at the top and the "after" image at the bottom of the page, with the same image size for both and made as large as possible yet still have both images fit on the page (i.e., within the margins). If you wish to show other figures for results you are free to do so, but it is not required. Of course, you can also have various figures as you deem appropriate in the introductory or background sections of your project report. You can show more than one "before" and "after" pair if you wish, to show variations on your compression approach, but it's not required.

For each "before" and "after" pair of images, you must also include in your project report results section (**in table format**) the following items:

- compression ratio (C) based on before/after comparisons of disk file size (including overhead),
- average bits/pixel based on the size of just the compressed image data size as reported by the MATLAB `whos` command (which excludes overhead),
- MSE (as just a number) of the decompressed image compared to the original,
- PSNR (expressed in decibels) of the decompressed image compared to the original, and
- a qualitative assessment made by you of how "good" the decompressed image looks.

Note that the first order estimate of Shannon entropy of the original image of Lena as provided to you is 7.4451 bits/pixel, a value you should include in your report in or near the table described above, to compare to your average bits/pixel results. The compression ratio will be calculated from the ratio of the "original" file size to the "compressed" file size (including overhead), when both are in `*.MAT -V6` format. For example, if your compressed image MAT file is half the size of the uncompressed image MAT file, you have achieved a $C = 2$. **Note:** Windows Explorer is rather imprecise regarding how it reports file size. To make sure everyone does it the same way, from Windows Explorer right click on the file name, select "Properties," and use the value shown as "Size" in bytes (do not use the "Size on disk" value as it varies according to how the hard disk was formatted). Include the version of Windows you are using.

In addition to the project scoring rubric with which you are already familiar, the "quality" of your compression results will be determined both objectively (i.e., quantitatively) and subjectively (i.e., qualitatively). For the quantitative measure, a large value of C along with a small value of MSE (and a small value of MSE means a relatively large value of PSNR, right?) is what you seek.[3] **Note that a value of $C \leq 1.0$ for your results is unacceptable**; you must achieve *some* level of compression! But since MSE and PSNR are not always a reliable measure of how "good" a compressed → decompressed image "looks" to the human eye, judging your results must also require

[3] Students often ask for a target value of C and PSNR; I hesitate to give you one as I don't want to constrain your efforts. However, past students have typically achieved $C \geq 5$ with a PSNR $\geq$ 30 dB using lossy methods.

somewhat of a qualitative assessment by you when you discuss results in your report, and by me as I score your project.

Questions/Discussion: The write-up for this project report should be reasonably brief, using wording and formatting suitable for an IEEE technical journal.[4] It should include some general background of pertinent concepts such as lossless versus lossy compression, Shannon entropy of data, and any key concepts your code uses (e.g., variable length coding, transform coding). Explain succinctly but clearly what algorithm(s) are performed by your programs, which if any Toolbox or other programs you used or that you "borrowed," why you chose the compression method you did, and what salient lessons you learned in the process of creating your programs. Finally, and most importantly, you should discuss intelligently your own overall interpretation of the quality of your results. This can appear in the Results section or in a separate Discussion section, as you prefer. Note that the Conclusions section only summarizes and ties together previously stated points in the report; *no new information* should appear in the Conclusions section.

Turn in: Turn in your project report in PDF file format, along with any original m-files you wrote as part of this project as separate files, via e-mail attachments. Name your PDF file "`Last1_Last2_proj04.pdf`" please, where "Last1" is the last name of team member 1, and "Last2" is the last name of team member 2. You must *also* include, as a minimum, your "best" `*.MAT` file that results from your compression program.[5]

Don't wait until the due date approaches to start this project! This project, similar to those before it, needs "sink-in" time in your brain, time to try a few different approaches, time to think about the results, and time to write (and re-write) the report for it to have the most benefit. Plan on the need to try several different ideas and techniques for your compression program before you get results you're willing to turn in, and *allocate sufficient time* to write a good report about it. Yes, it's the end of the semester, and it should be no surprise to you that you'll have other assignments coming due for other classes! *Plan your work and work the plan...*

[4] Your team's project report must follow the format referenced on the course web site. Be sure that the name of each team member appears on the first page of the report (below the title of the report). Be sure to use proper grammar and logical, organized sentences. **Any equations in your report should be typeset with an equation editor** (or use the math mode of LaTeX). Do *not* copy and paste any equations from the course lesson slides into your report—that method always looks unprofessional. All figures or tables should have descriptive captions.

[5] You'll probably have to rename your `*.MAT` file to have a different extension such as `*.DAT` to prevent an overly "helpful" Microsoft Outlook filter from blocking the attachment. Alternatively, you can include the `*.MAT` file as part of a `*.ZIP` or a `*.RAR` file. In any case, it's *your* responsibility to ensure the `*.MAT` file was delivered to me properly and on time.