**HƯỚNG DẪN CHẠY CHƯƠNG TRÌNH  
HaarWavelet**

1. **CÀI ĐẶT**

* Ứng dụng HaarWavelet có thể chạy trên nền Windows XP trở lên.
* Ứng dụng không cần cài đặt.
* Do ứng dụng được phát triển bằng Visual Studio cùng với OpenCV 2.48 nên để chạy ứng dụng cần 11 tập tin dll (đã để sẵn cùng với tập tin exe):
  + opencv\_calib3d248.dll
  + opencv\_contrib248.dll
  + opencv\_core248.dll
  + opencv\_features2d248.dll
  + opencv\_flann248.dll
  + opencv\_highgui248.dll
  + opencv\_imgproc248.dll
  + opencv\_legacy248.dll
  + opencv\_ml248.dll
  + opencv\_objdetect248.dll
  + opencv\_video248.dll

1. **CHỨC NĂNG**
2. Chức năng của ứng dụng

Biến đổi wavelet với tham số iteration.

Giảm nhiễu, với mức threshold.

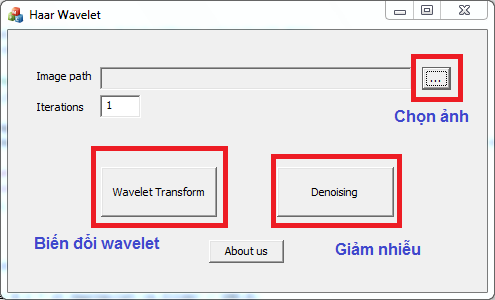
1. Một số lưu ý

Tập ảnh để chạy chương trình nên có kích thước chiều dài và chiều rộng là 2n (do thuật toán sử dụng để tạo wavelet giảm theo lũy thừa của 2).

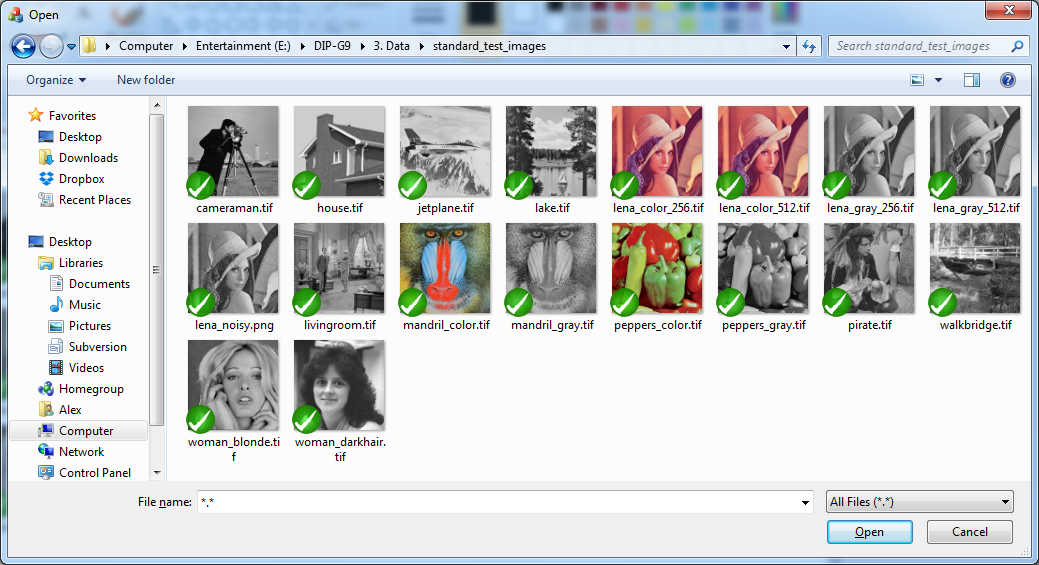
Ảnh kết quả của biến đổi wavelet nằm trong thư mục chứa ảnh ảnh gốc.

1. Giao diện sử dụng

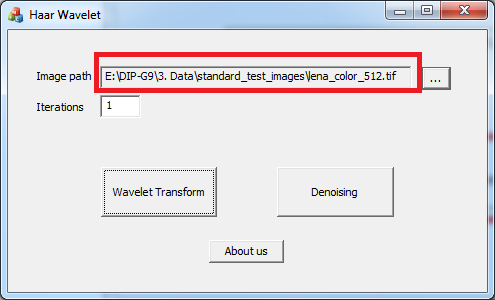
Khi chạy tập tin DemoHaarWavelet.exe sẽ mở màn hình giao diện chính.



Bước đầu tiên, người dùng cần chọn ảnh muốn biến đổi wavelet hoặc giảm nhiễu bằng biểu “…” ở góc trên bên phải.



Ví dụ, người dùng chọn ảnh “lena\_color\_512.tif” trong thư mục standard\_test\_images, sau đó nhấn vào nút Open. Ứng dụng sẽ trở lại giao diện chính. Đường dẫn đến tập tin ảnh vừa chọn sẽ được điển thị ở kế bên ô “Image path”.



Sau đó người dùng có thể chọn chức năng “Wavlet transform” đế biến đổi wavelet (với tham số Iteration tương ứng với cấp biến đổi) hoặc Denoising để giảm nhiễu.

Ví dụ biến đổi wavelet với iteration là 1. Ứng dụng sẽ hiển thị lên 2 cửa sổ như hình bên dưới. Cửa sổ FWT (Forward Wavelet Transform) là hình đã biến đổi bằng Haar Wavelet. Ảnh được thu nhỏ bằng phân nửa kích thước ảnh gốc (do iteration = 1). Cửa số IWT (Invert Wavelet Transform) là ảnh sau khi biến đổi bằng Haar Wavelet được biến đổi ngược trở lại với kích thước bằng với ảnh gốc.

Đồng thời 2 tập tin tương ứng là Forward\_1\_lena\_color\_512.tif và Invert\_1\_lena\_color\_512.tif cũng được tạo ra trong cùng thư mục chứa ảnh gốc.

|  |  |
| --- | --- |
|  |  |

Ví dụ người dùng chọn iteration là 2 với cùng ảnh trên thì kết quả như bên dưới.

|  |  |
| --- | --- |
|  |  |

Ví dụ người dùng chọn ảnh “lena\_noisy.png” và chọn chứng năng “Denoising”. Chương trình sẽ mở một cửa số để người dùng chọn mức Threshold khử nhiễu. Threshold càng lớn thì càng giảm nhiễu, nhưng lớn quá thì sẽ bị blocky.

Ảnh với threshold = 0, bị nhiễu khá nhiều như hình dưới.



Ảnh sau khi điều chỉnh threshold = 50 như hình bên dưới.



Nếu chọn threshold = 100, ảnh sẽ khử nhiễu hoàn toàn nhưng bị hiện tượng blocky như hình dưới.

