

```

142 %      |      |
143 %      S---|      |--HPy{N}
144 %      |      |
145 %      |--LP--- ... ---|
146 %      |      |
147 %      |--LP
148 %
149 %      S - Signal to be decomposed.
150 %      N - Number of channels to compute.
151 %      D - Downsample LP-component between channels.
152
153 % Filter coefficients for analyzing filters.
154 hx = [0.0625 0.25 0.375 0.25 0.0625];
155 gx = [0 1 -1];
156
157 lpChannel = img;
158
159 if(sampl)
160     for iChannel = 1:numChannels
161         hpChannels{iChannel}(:, :, 1) = mxSimpleDiff(lp
162             hpChannels{iChannel}(:, :, 2) = mxSimpleDiff(lp
163
164         lpChannel = imfilter(imfilter(lpChannel, hx, 'r
165             hx', 'replicate', 'conv'));
166         lpChannel = lpChannel(1:2:end, 1:2:end);
167     end
168 else
169     for iChannel = 1:numChannels
170         hpChannels{iChannel}(:, :, 1) = mxSimpleDiff(lp
171             2^(iChannel-1)-1);
172         hpChannels{iChannel}(:, :, 2) = mxSimpleDiff(lp
173             2^(iChannel-1)-1)';
174
175         lpChannel = imfilter(imfilter(lpChannel, zeroP
176             'replicate', 'conv'), zeroPad(hx, iChannel)
177     end
178 end
179 % -----
180
181 % ----- IDWT -----
182 function img = idwt(hpChannels, lpChannel, sampl)
183 %IDWT Reconstructs a signal that has been decomposed
184 %      S = IDWT(HPCHANNELS, LPCHANNEL, SAMPL)
185 %      HPCHANNELS, LPCHANNEL - Output from DWT, write 'h
186 %      SAMPL - Boolean that indicates we
187 %      downsamped during decomp
188
189 % Filter coefficients for reconstructing filters.
190 hx = [0.0625 0.25 0.375 0.25 0.0625];
191 kx = [-0.00390625 -0.03515625 -0.14453125 -0.36328125
192     0.14453125 0.03515625 0.00390625 0];
193 lx = [0.001953125 0.015625 0.0546875 0.109375 0.63671
194     0.0546875 0.015625 0.001953125];
195
196 % Split signal into channels with or without downsamp
197 % channels.
198 iChannel = length(hpChannels);

```