Fig. 6.2: **Image similarity via ensemble clustering.** Different trees (whose leaves are denoted by different colour curves) induce different image partitions. The red tree yields the partition $\{\{a,b,c,d\},\{e,f\},\{g,h\}\}$. The green tree yields the partition $\{\{a,b,c\},\{d,e,f\},\{g,h\}\}$. The overlap between clusters in different trees is captured mathematically by the forest affinity matrix W. In W we will have that image e is closer to image c than to image g. Therefore, ensemble-based clustering induces data affinity. See text for details.

Computational efficiency. In this algorithm the bottleneck is the solution of the eigen-system (6.7) which could be slow for a large number of input points k. However, in (6.9) only the $d' << k$ bottom eigenvectors are necessary. This, in conjunction with the fact that the matrix L is usually very sparse (especially for the binary affinity model) can yield efficient implementations. Please note that only one eigen-system