## Tracking of graph branches

* For each vertex in G1, find a corresponding vertex in G2.
* Let u and v in G1 and u’ and v’ are their correspondences in G2. If u and v are neighbors, then u’ and v’ should be neighbors as well.
* If we can establish good tracking, we can use the motion to improve grouping of vertices into separate objects.
* Motion should also be a part of the MST construction.
* If we can come up with **a good correspondence measure based on the graph structure**, then we can derive the motion vector and use it to derive the MST.
* The good measure should include: proximity,
* How to deal with edges whose motions are collinear.

## How to utilize the branches and their motions

* We can group branches based on their motions.
* How can we incorporate more complex motions? Articulated parts?
* How can we use the grouping to region partitioning?
* Once we can establish proper grouping of moving branches, we can probably draw partition via the space filling technique.

## Can we approach the problem as graph-cut?

* To compare b1 and b2 in the same frame, we should compare their ascendants back by a few frames.

## Can we encode more complex motions (rotation, shear, etc.)?



## Unit testing motion detection, extraction, grouping.

* Need a simple test image with ground truth. => Use Processing?