Topological Data Analysis: Understanding Optical Flow

IMA Short Course June 2009

Henry Adams, Gunnar Carlsson, Guillermo Sapiro

(An exercise before we begin)

- What space do you get when you identify antipodal points of a torus?

  $S^1 x S^1/(\alpha,\beta)\approx(-\alpha,-\beta)$

Goal:

Make topological data analysis research real

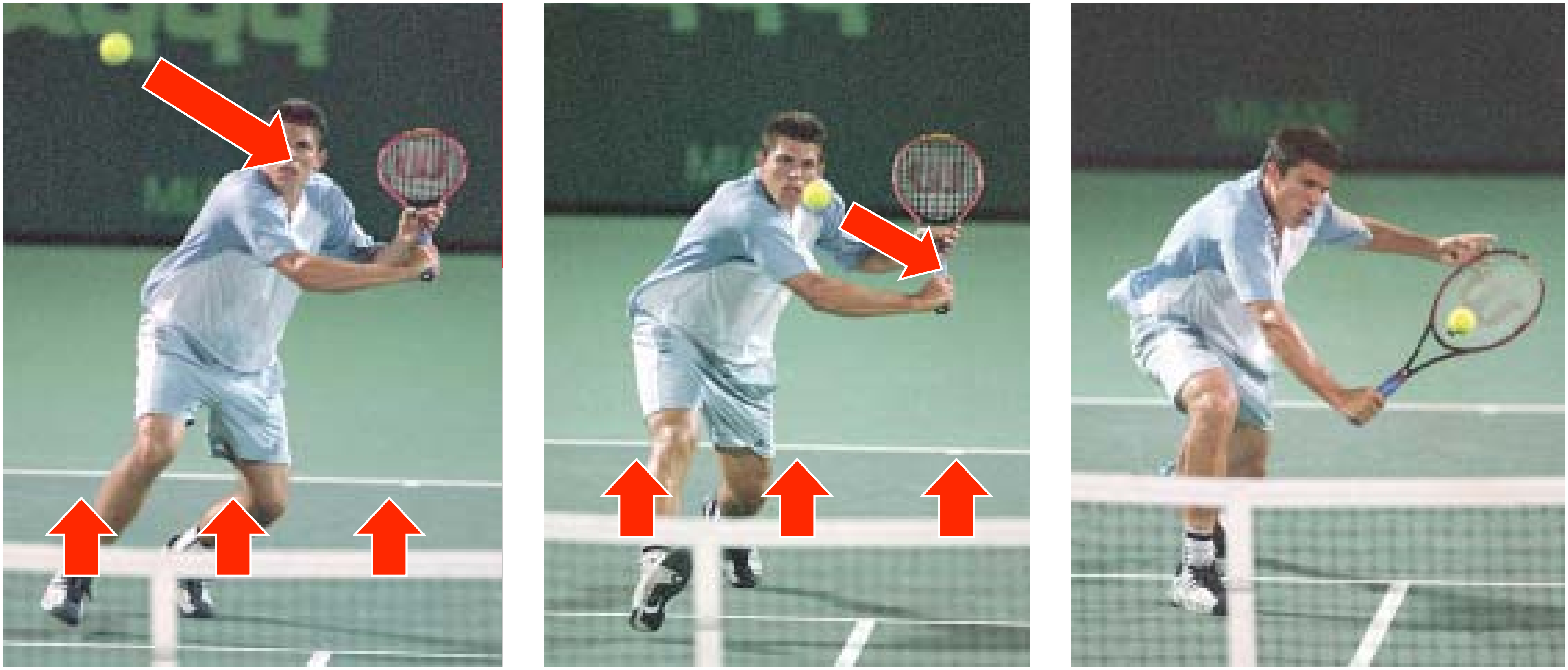
Lesson:

Use k-skeleton to guess $(k+1)$-skeleton

Contents

- What is optical flow?
- The database
- Preprocessing
- Topological results
- Zigzag persistence example

What is optical flow?



- Optical flow is the apparent motion (or projected motion) in a video.
- Vector field.
- Sources are scene motion and camera motion.

Who cares?

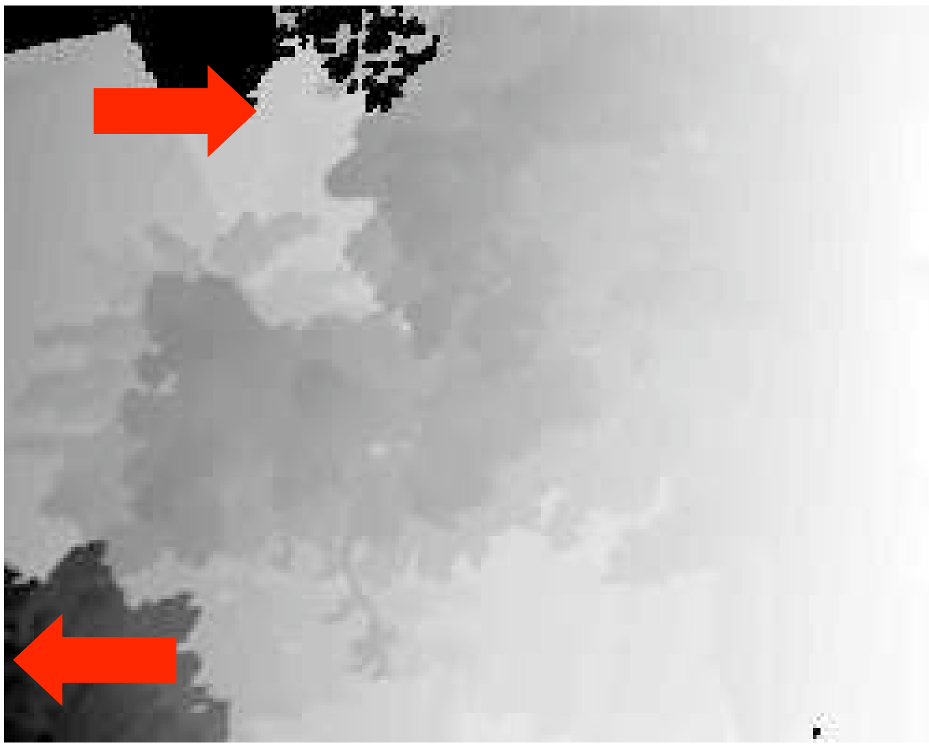
- Optical flow estimation is used for
  - Object segmentation and tracking
  - Video compression
  - Robot navigation.
- Estimation ill-posed.
- A better understanding of optical flow allows better estimation

Optical flow database

- Stefan Roth and Michael J. Black create 800 *natural* ground truth optical flow images.
- Pair range images with camera motions.
- No scene motion.

Optical flow database

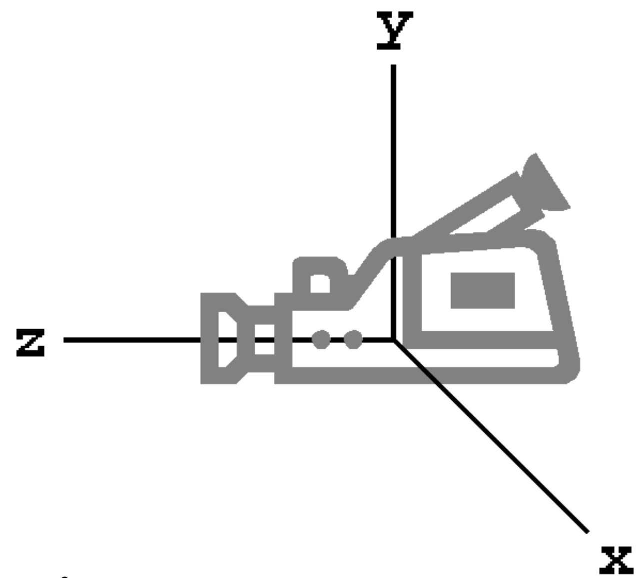
Horizontal component **Vertical component**



- White pixels have positive flow, black pixels negative.

Component databases

Camera motion properties



- Six camera motions.
- Hand-held and car-mounted cameras.
  - more $\pm x$ translation than $\pm y$ translation.
  - more $+z$ translation than $-z$ translation.
- Three rotations produce low-contrast flow (away from pole).

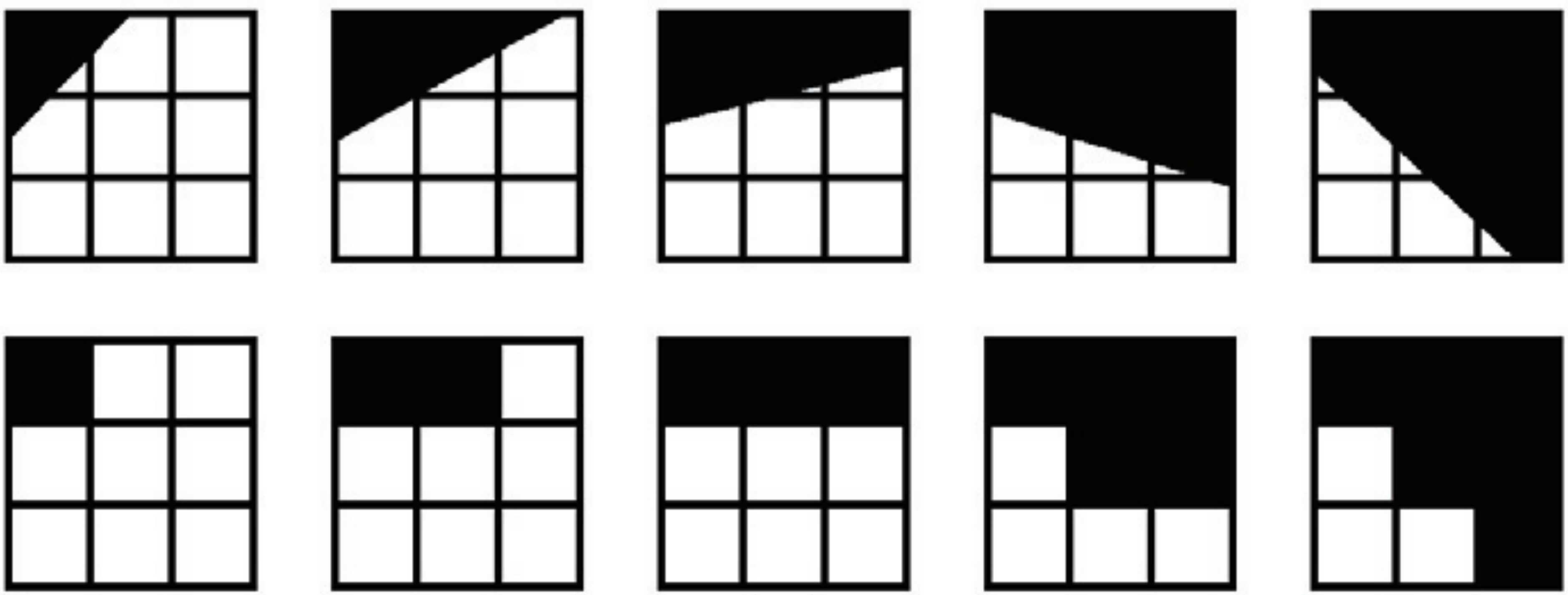
Range image properties

- Roth and Black use the range image database of Mumford et al.
- Mumford et al. took the top 20% highest-contrast range 3x3 patches and normalized them.
- Found clusters near binary patches but no geometric overview.

Sample range images

Range image properties

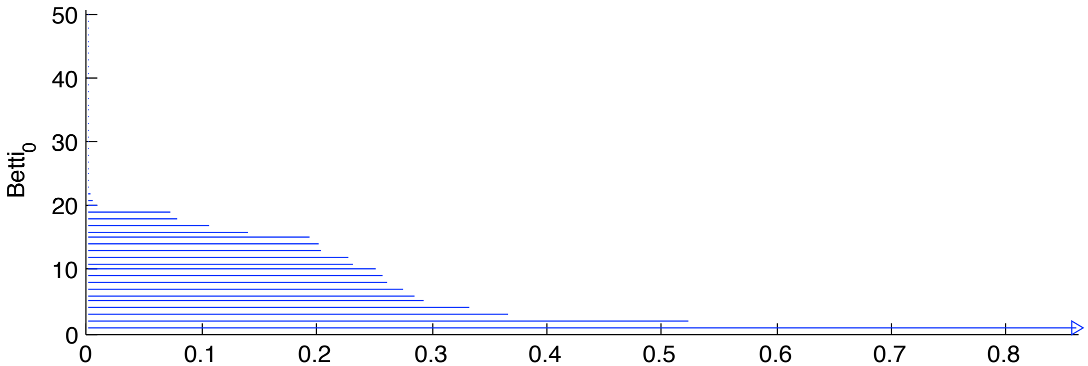
- Why binary patches?
  1. Hitting two objects gives only two values.
  2. Sub-resolution detail not averaged

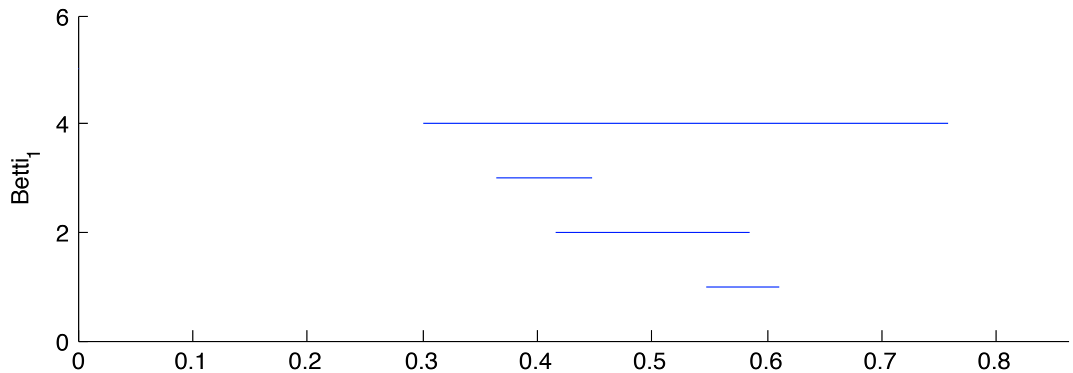


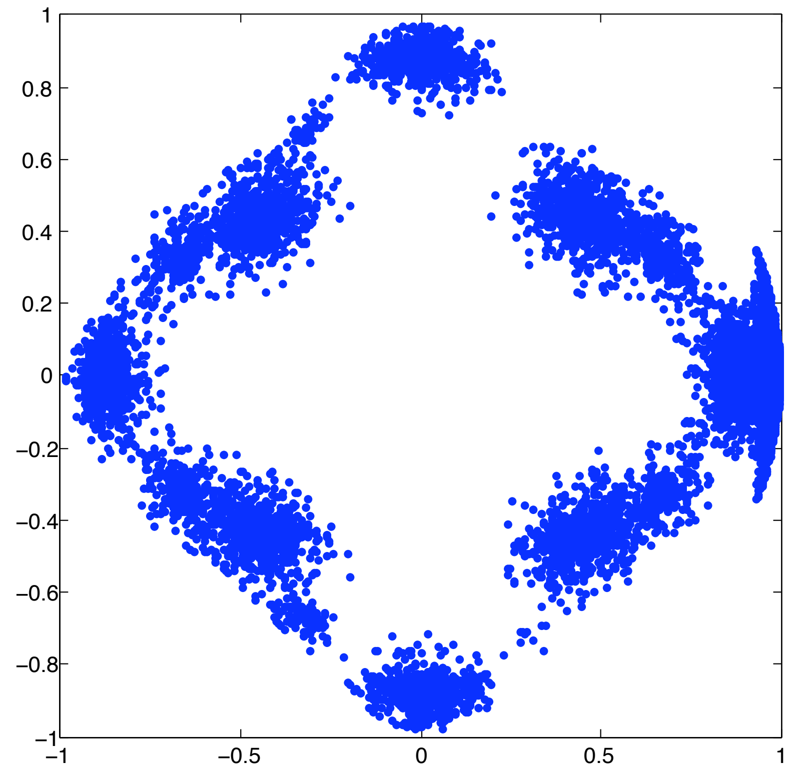
Range image properties

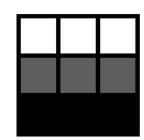
- We had previously studied these range images.
- Same preprocessing as Mumford et al., then we took dense core subsets.

3x3 range patches

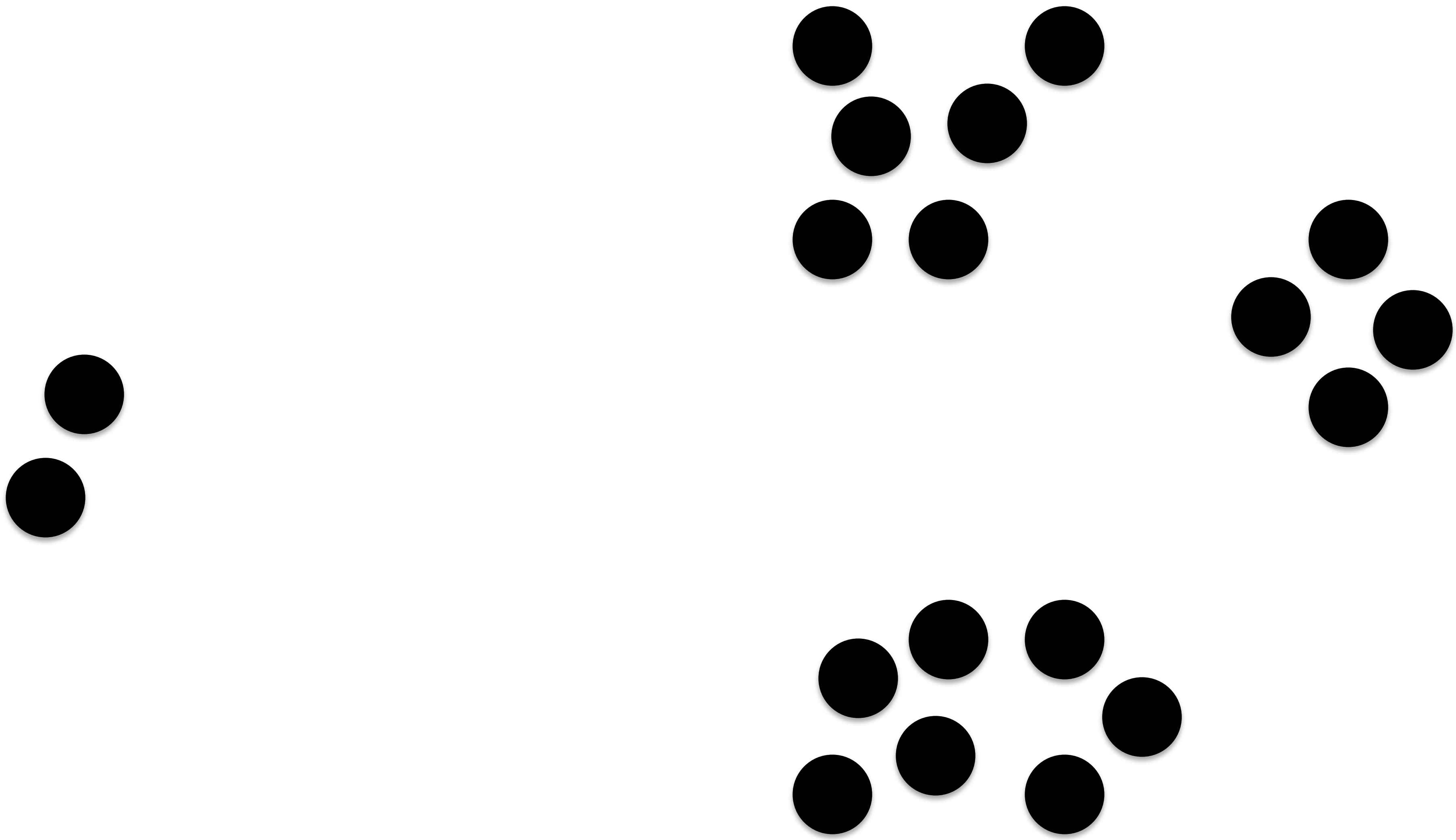




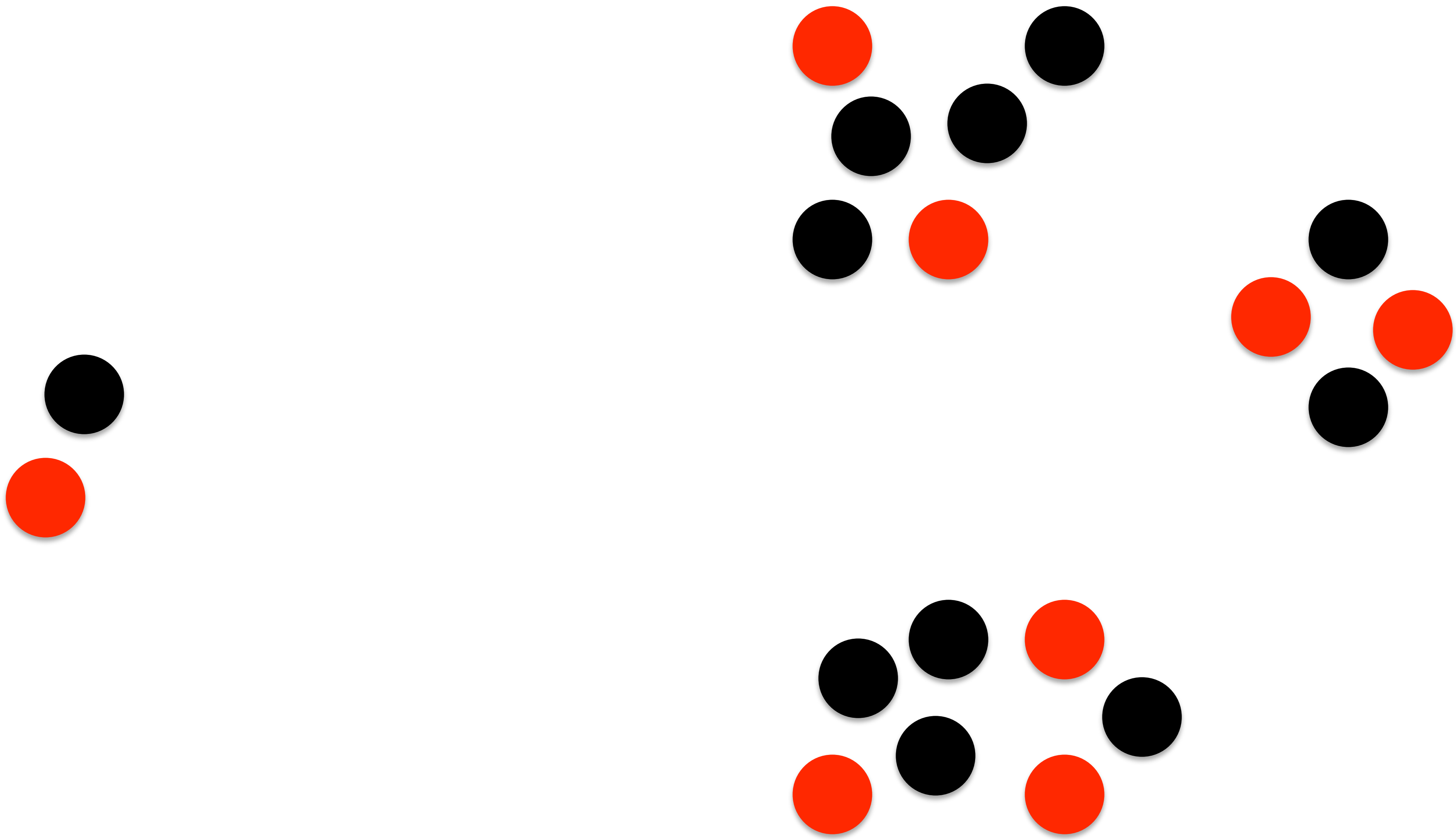


- ≈12 clusters in the shape of a circle.
- Clusters centered near binary patch (or )
- 3x3 binary patches too course to encode circle.

Trouble with W and LW_2

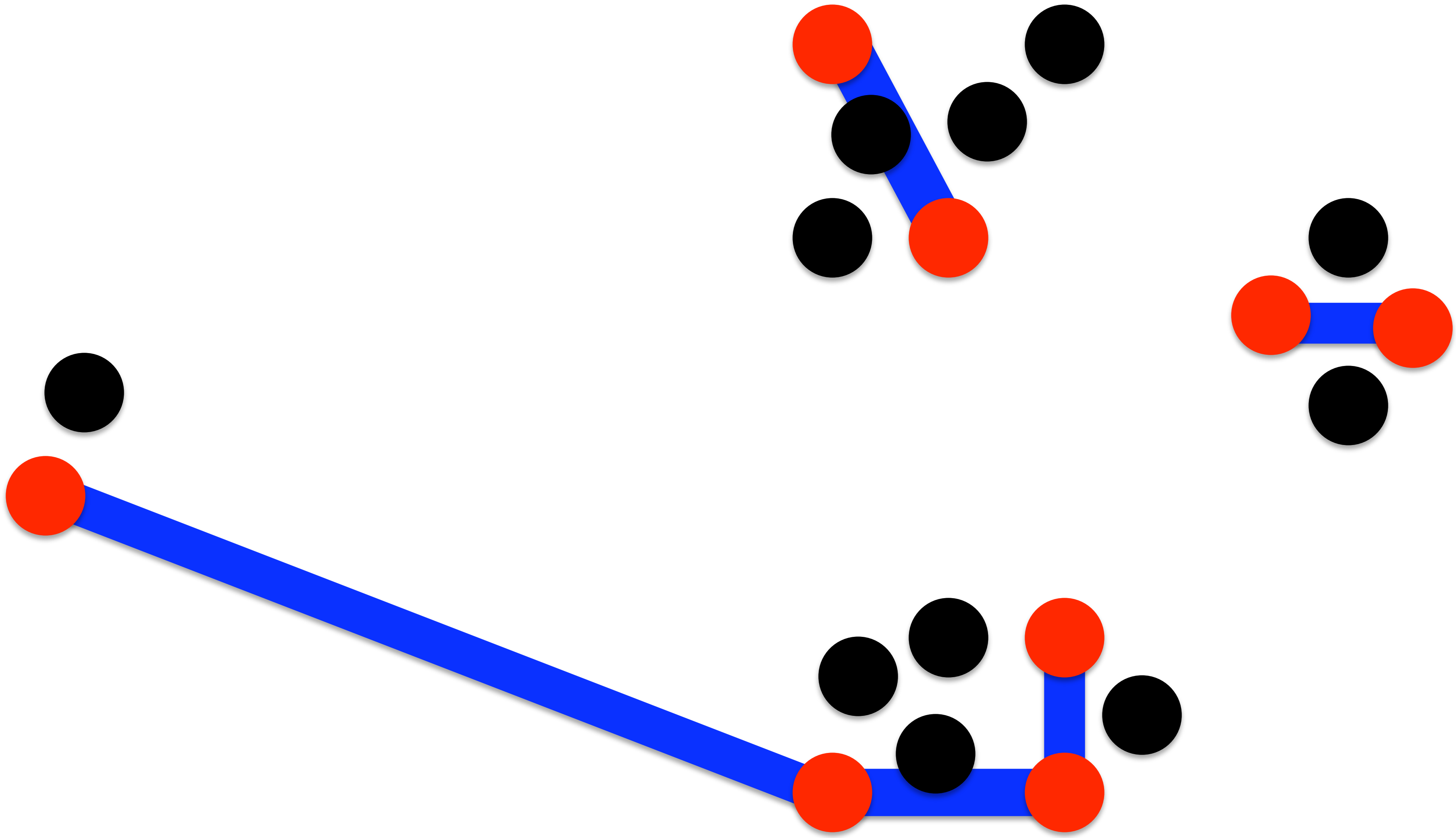


Trouble with W and LW_2



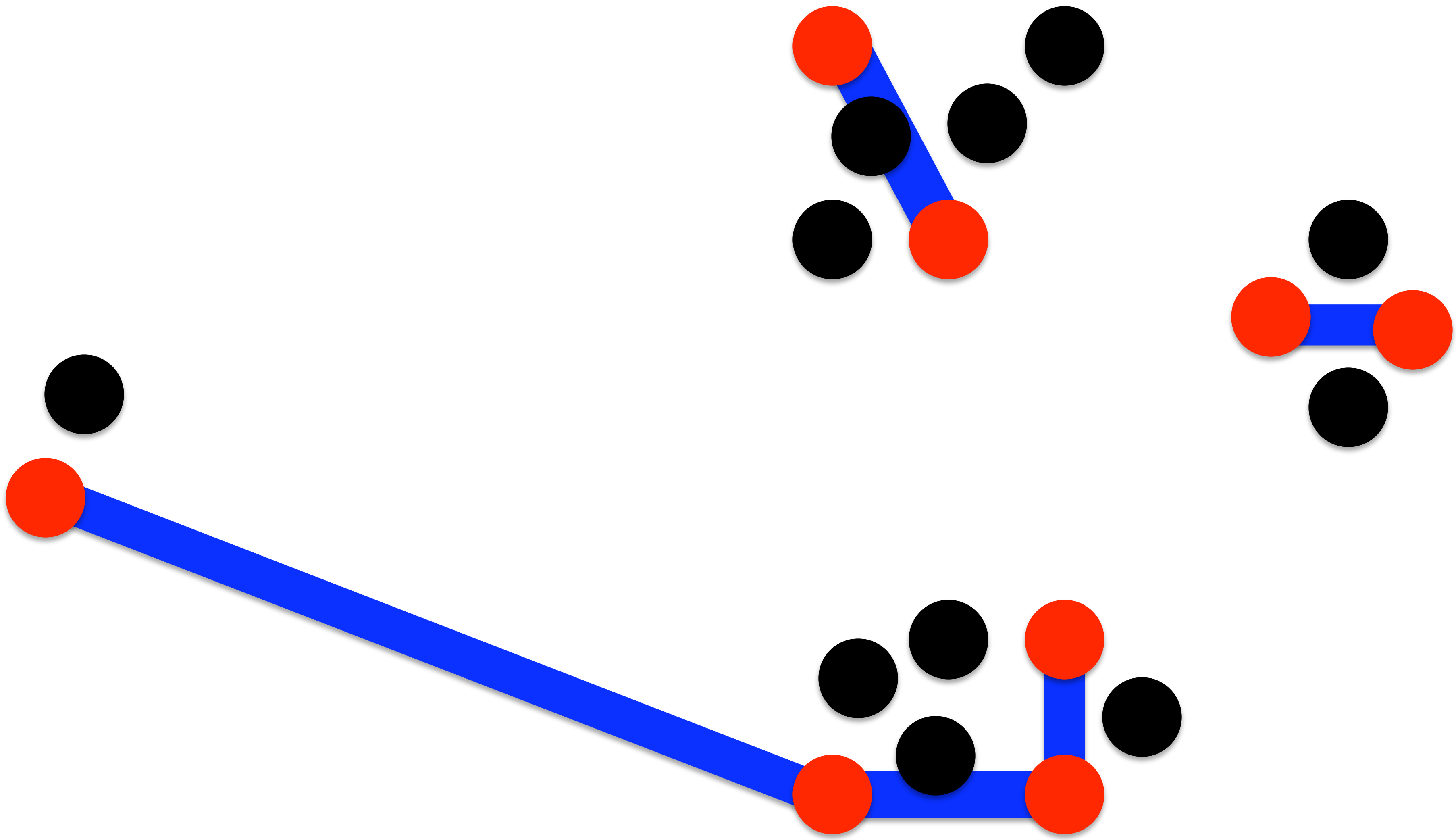
Landmark vertices

Trouble with W and LW_2



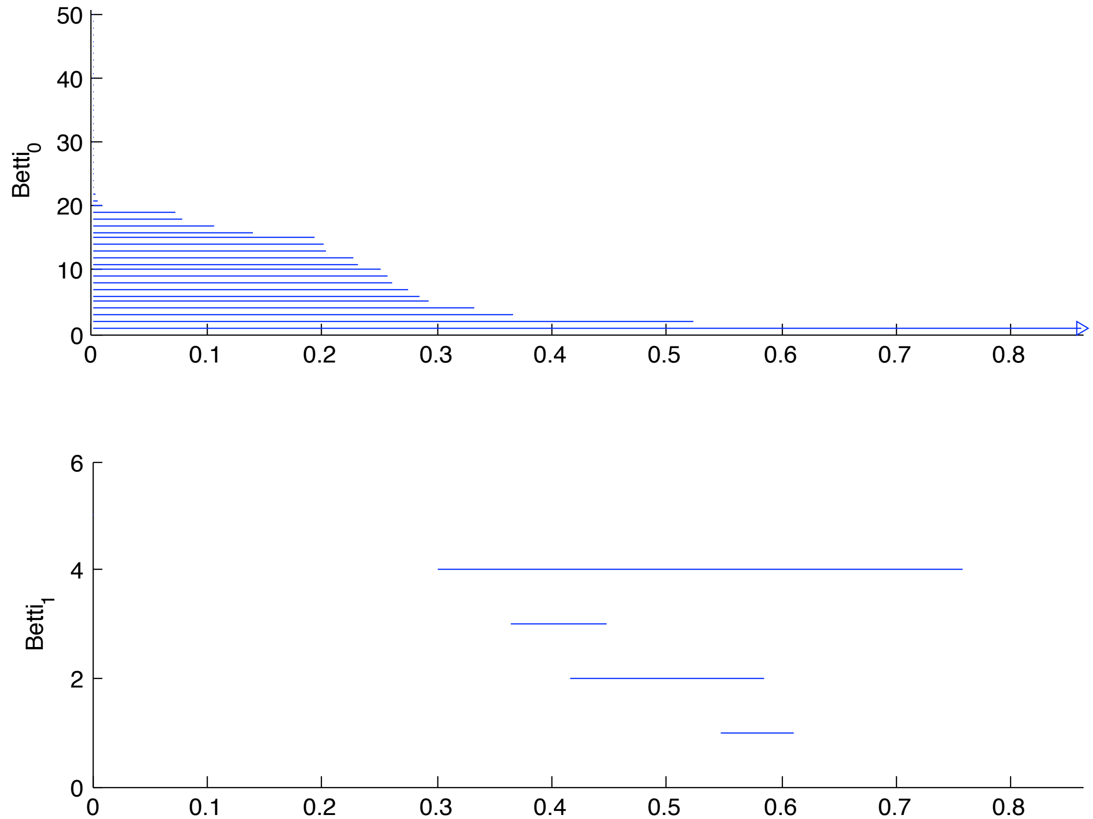
Edges at filtration time t=0

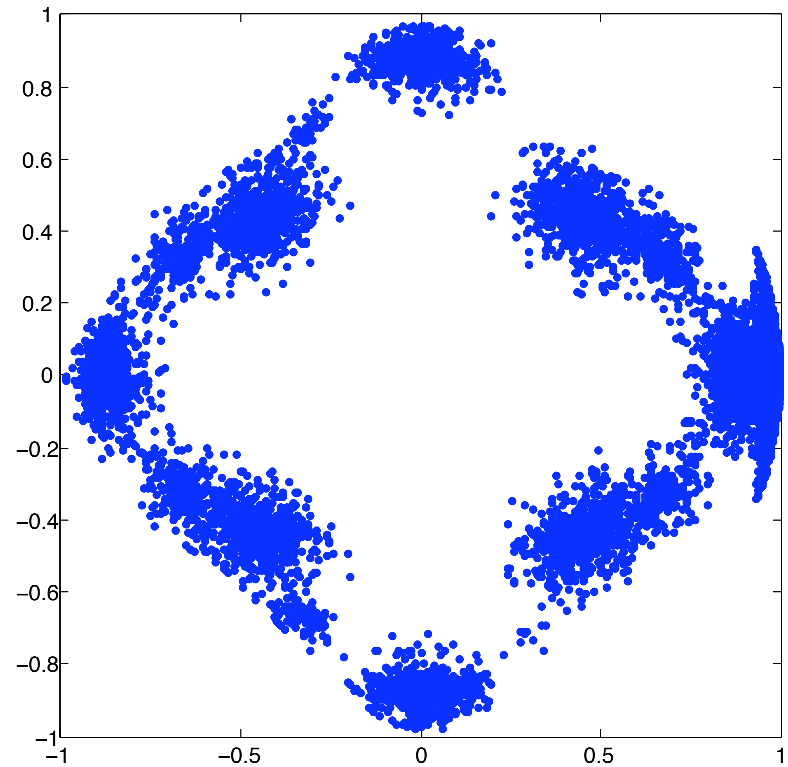
Trouble with W and LW_2

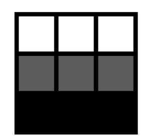


(Not a problem with LW_0 or LW_1)

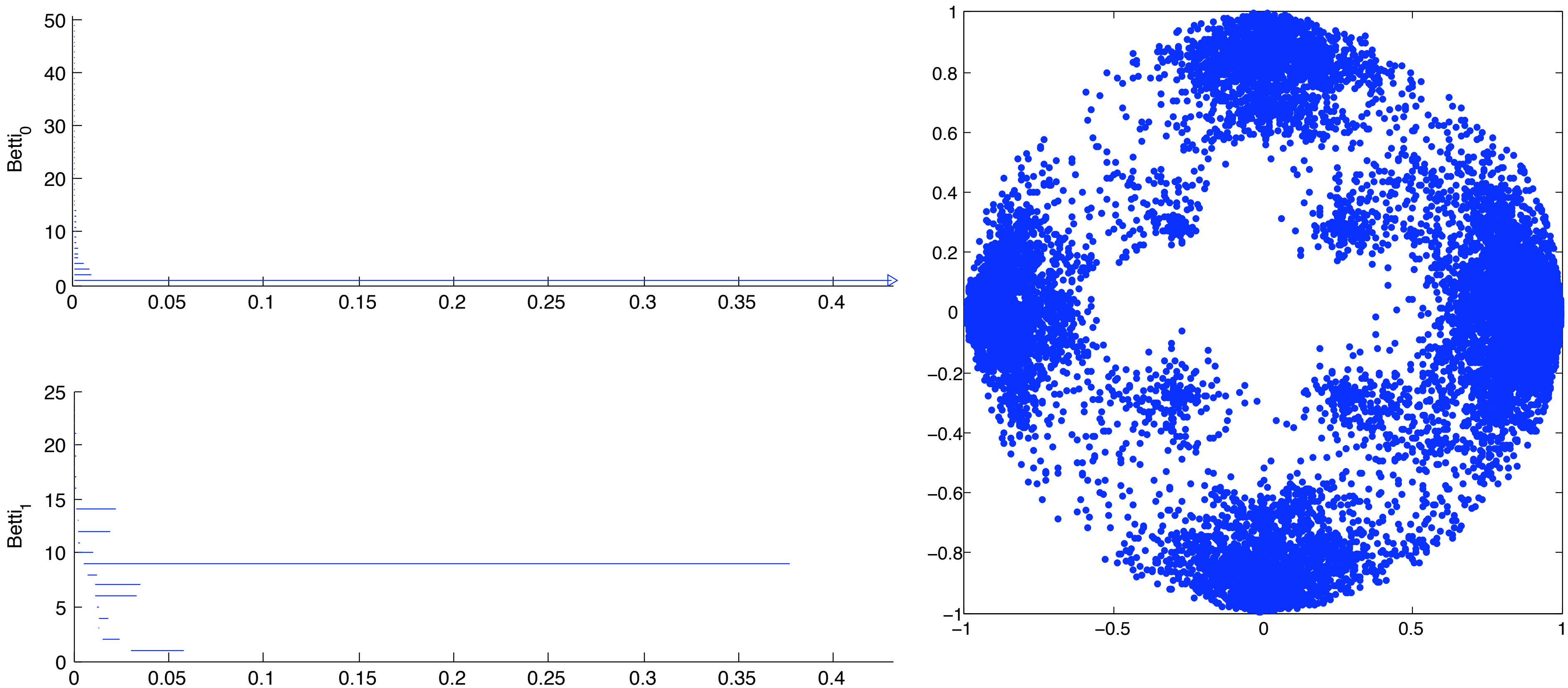
3x3 range patches





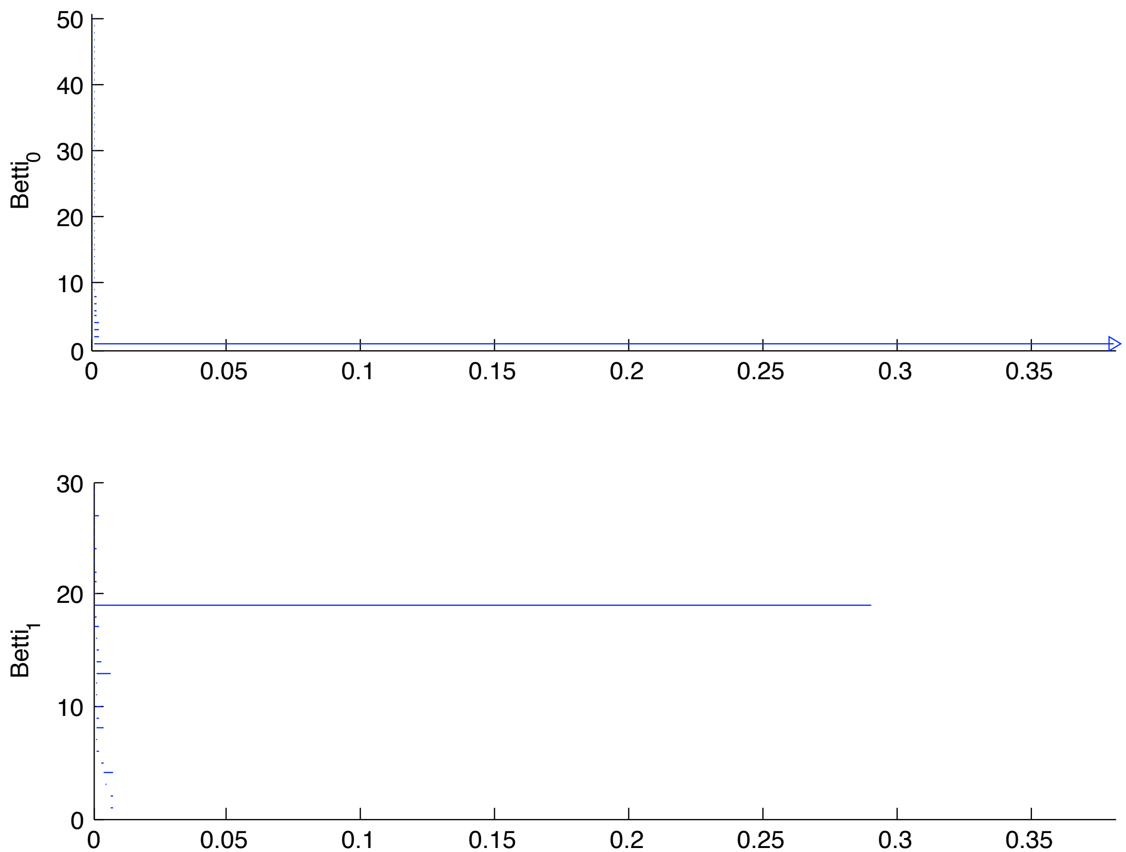
- ≈12 clusters in the shape of a circle.
- Clusters centered near binary patch (or )
- 3x3 binary patches too course to encode circle.

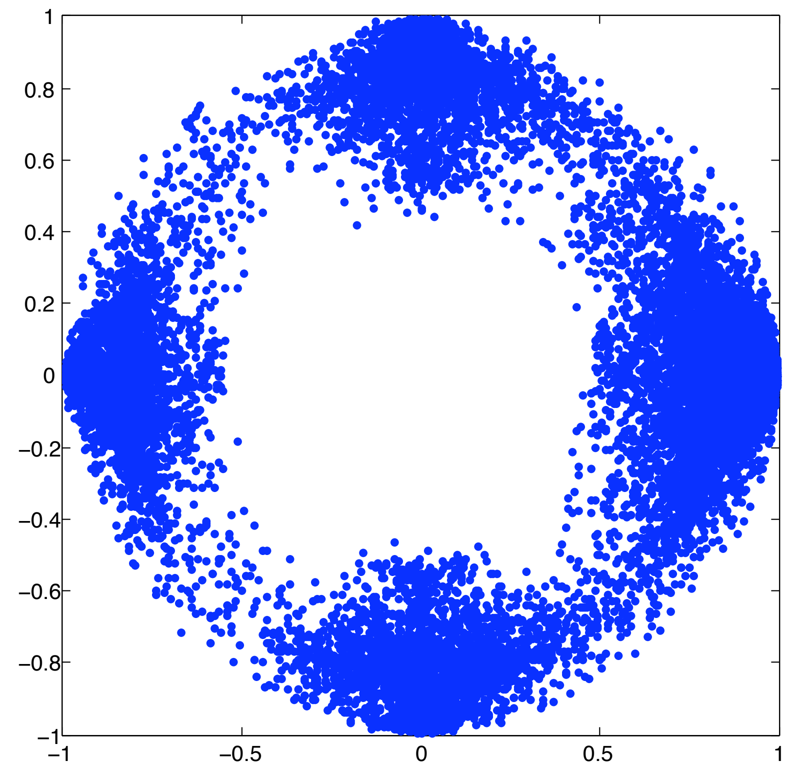
5x5 range patches



- Get a circle.

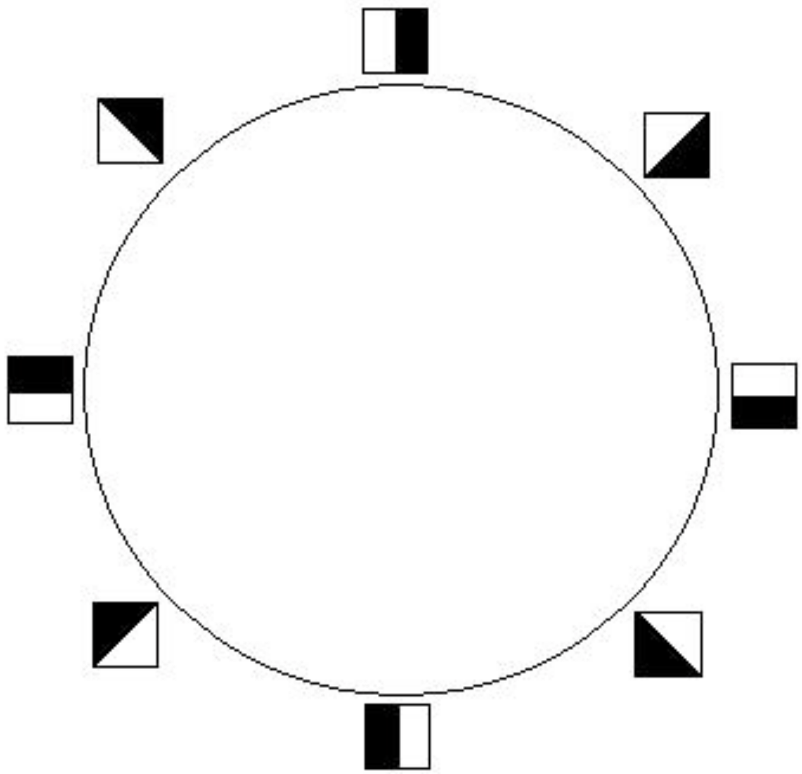
7x7 range patches

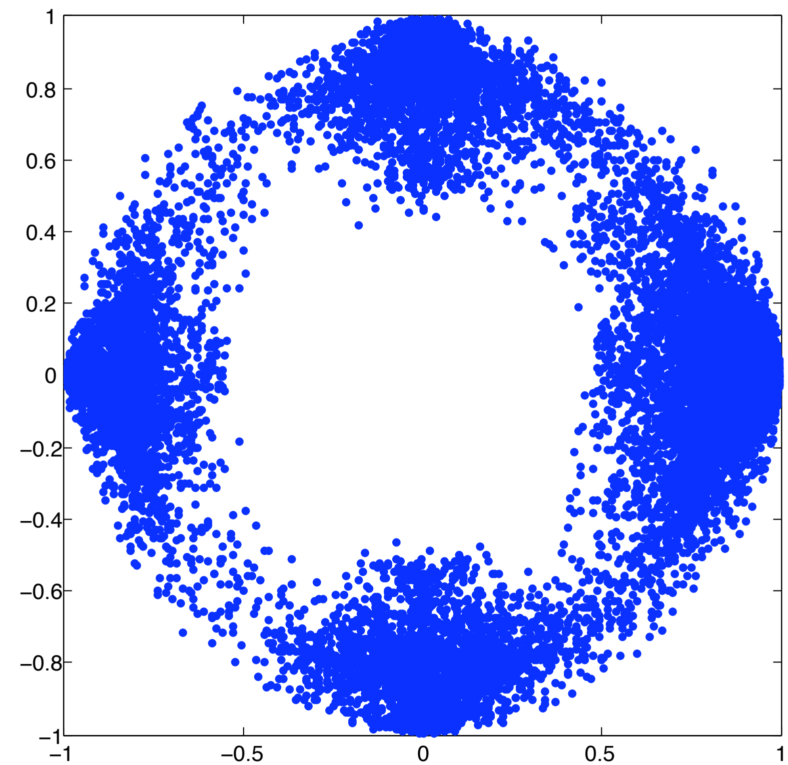




- Get a circle.

Range primary circle



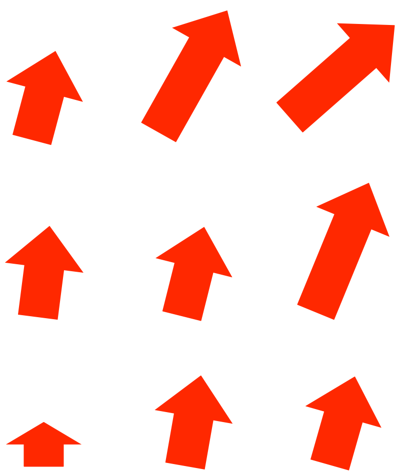


- Horizontal and vertical edges most common.
- No secondary circles appear

Optical flow

Preprocessing

1. We start with 250,000 random 3x3 patches from the Roth and Black optical flow database.



$$\begin{bmatrix} (u_1, v_1) & (u_4, v_4) & (u_7, v_7) \\ (u_2, v_2) & (u_5, v_5) & (u_8, v_8) \\ (u_3, v_3) & (u_6, v_6) & (u_9, v_9) \end{bmatrix}$$

Each patch x is a point in $\mathbb{R}^{18}$.

Preprocessing

2. Select the top 20% high-contrast patches.

$$\|x\|_{D_{18}} = \sqrt{\sum_{i \sim j} \|(u_i, v_i) - (u_j, v_j)\|^2}$$

3. Normalize each patch to have zero average flow and contrast norm one.

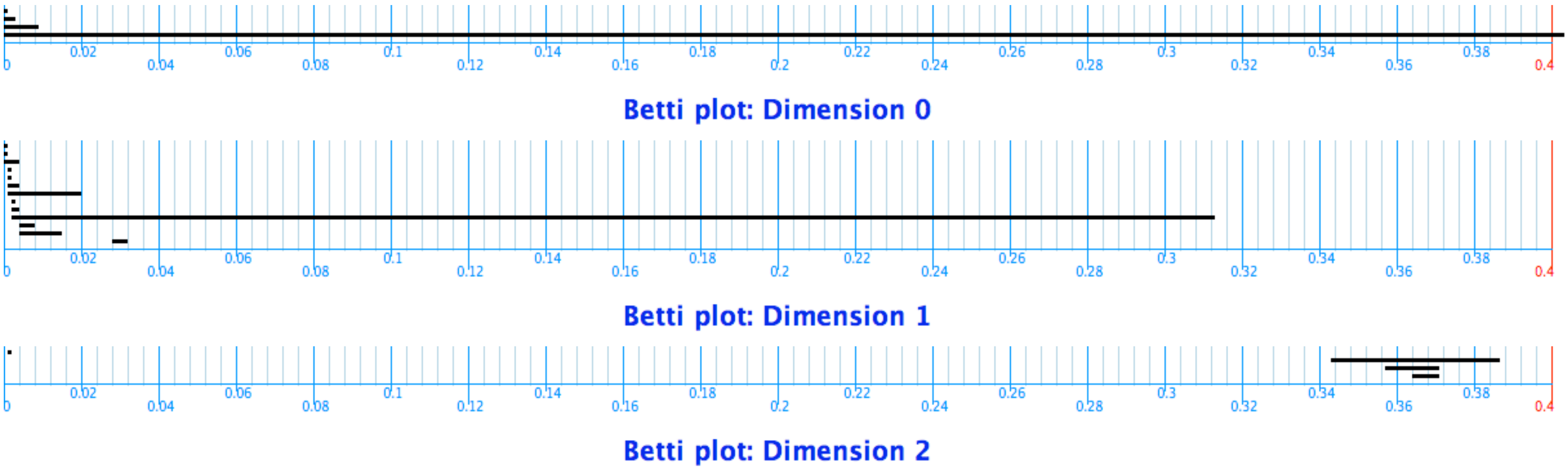
Ignoring structure is sometimes helpful.

We get 50,000 high-contrast, normalized, optical flow patches on a 15-dimensional ellipsoid in $\mathbb{R}^{18}$.

Preprocessing

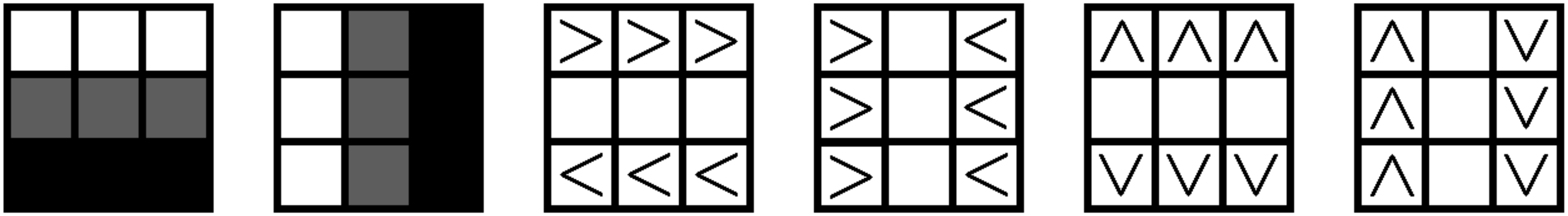
4. Restrict to dense core subsets:
   - Let $\boldsymbol{\rho}_k$ be the distance from patch x to its k-th nearest neighbor.
   - Let $X(k,p)$ be the top p% densest points, using density estimator $1/\boldsymbol{\rho}_k$.
   - We analyze core subsets $X(300,30)$ and $X(15,30)$.

X(300,30)



- We get $\text{Betti}_0 = \text{Betti}_1 = 1$, the barcode of a circle.

DCT basis vectors



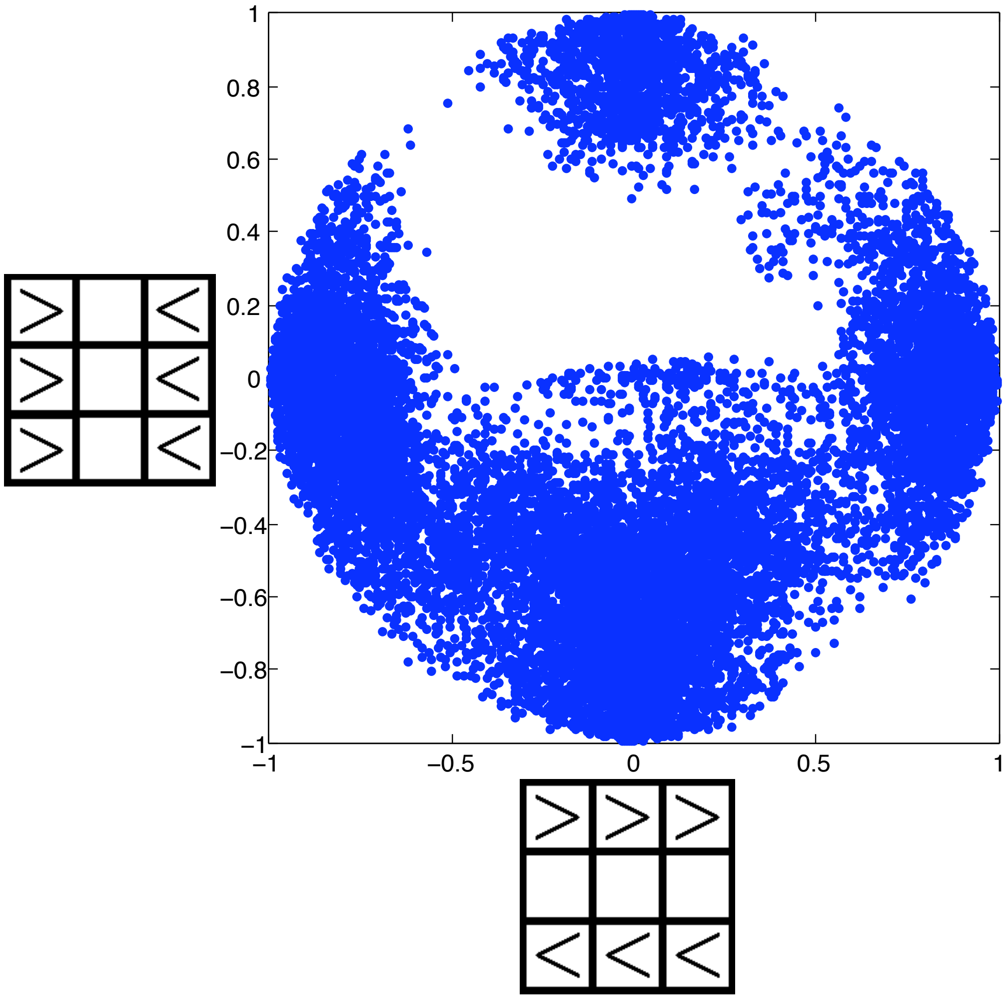
e1 e2 e1u e2u e1v e2v

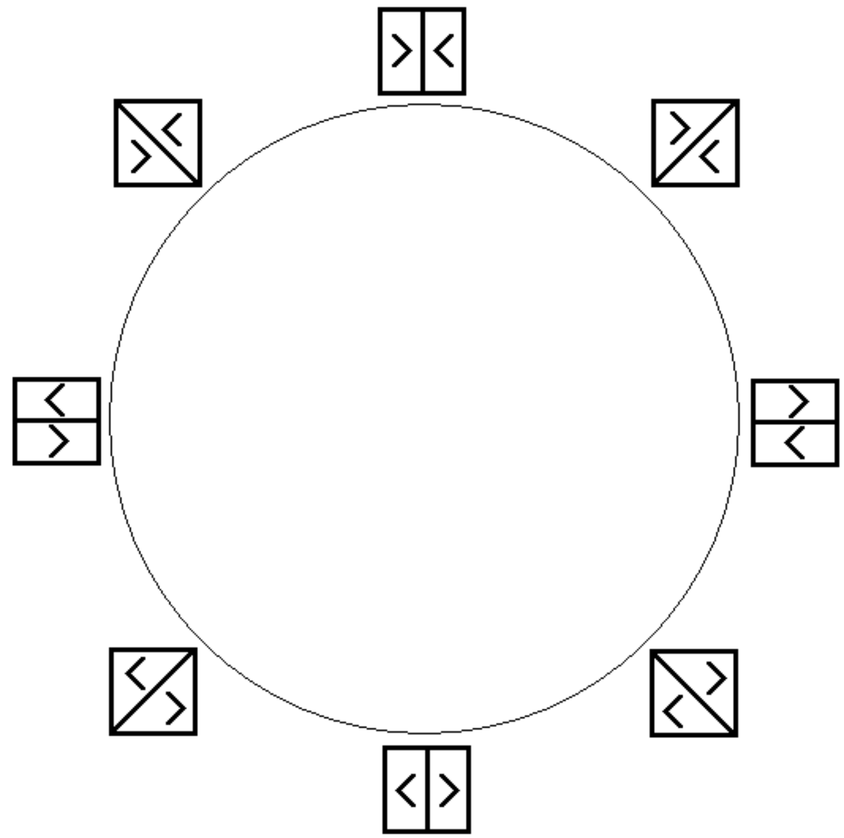
Linear gradient basis vectors, made into horizontal and vertical flow patches

- Preview: think of e1 as a range patch, with black foreground, white background.
- Rightward (+*x*) camera translation over e1, normalized, produces e1u.

X(300,30)

- Project onto e1u and e2u.
- Remember more $\pm x$ than $\pm y$, and more $+z$ than $-z$ camera translation?

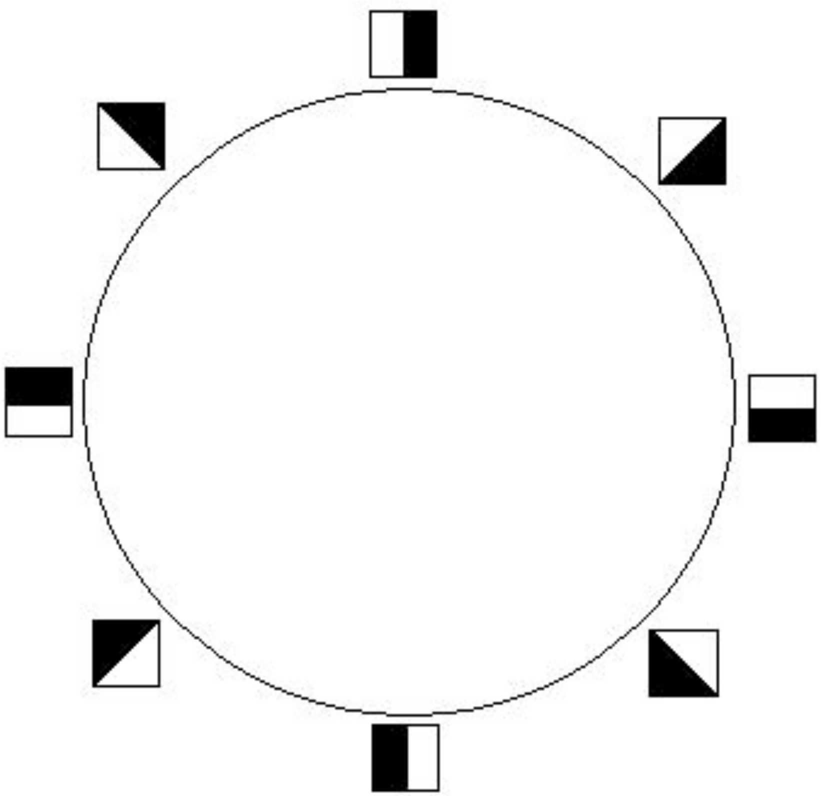




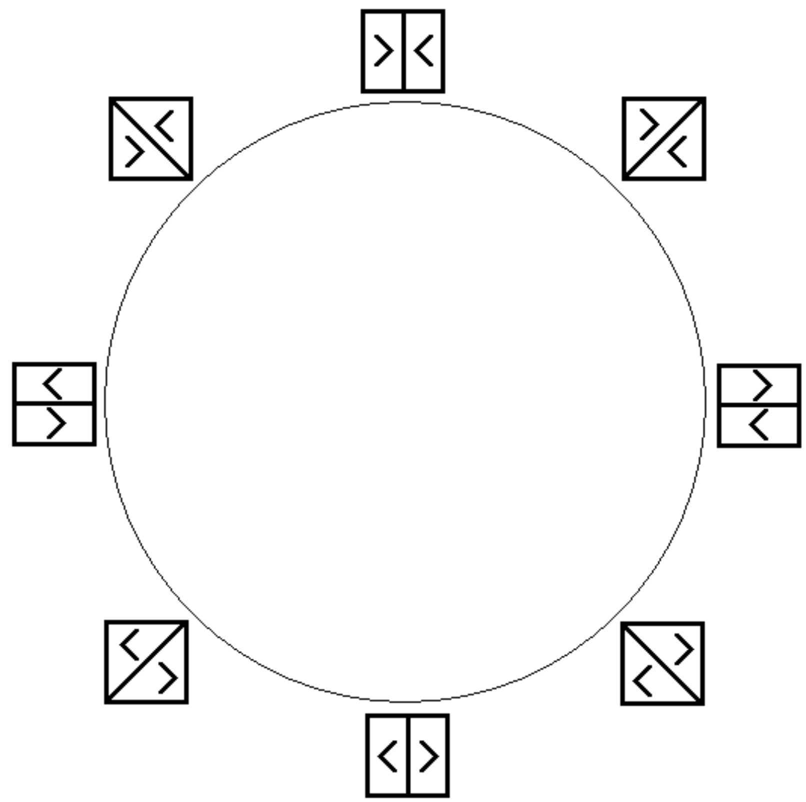
Horizontal flow circle

Source for horizontal flow circle?

- Combine most common elements from each component database: apply $\pm x$ camera translation to range primary circle.

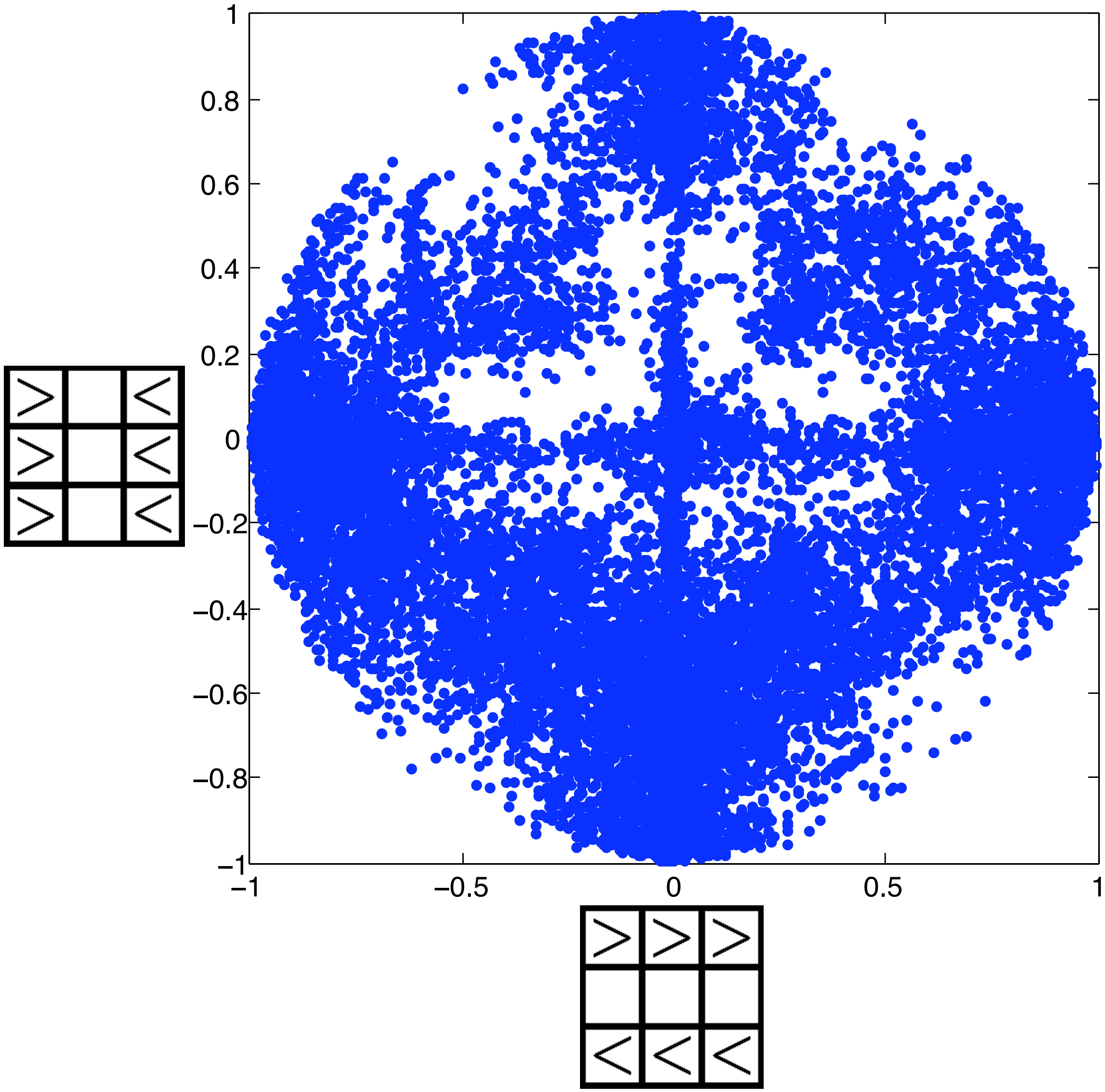


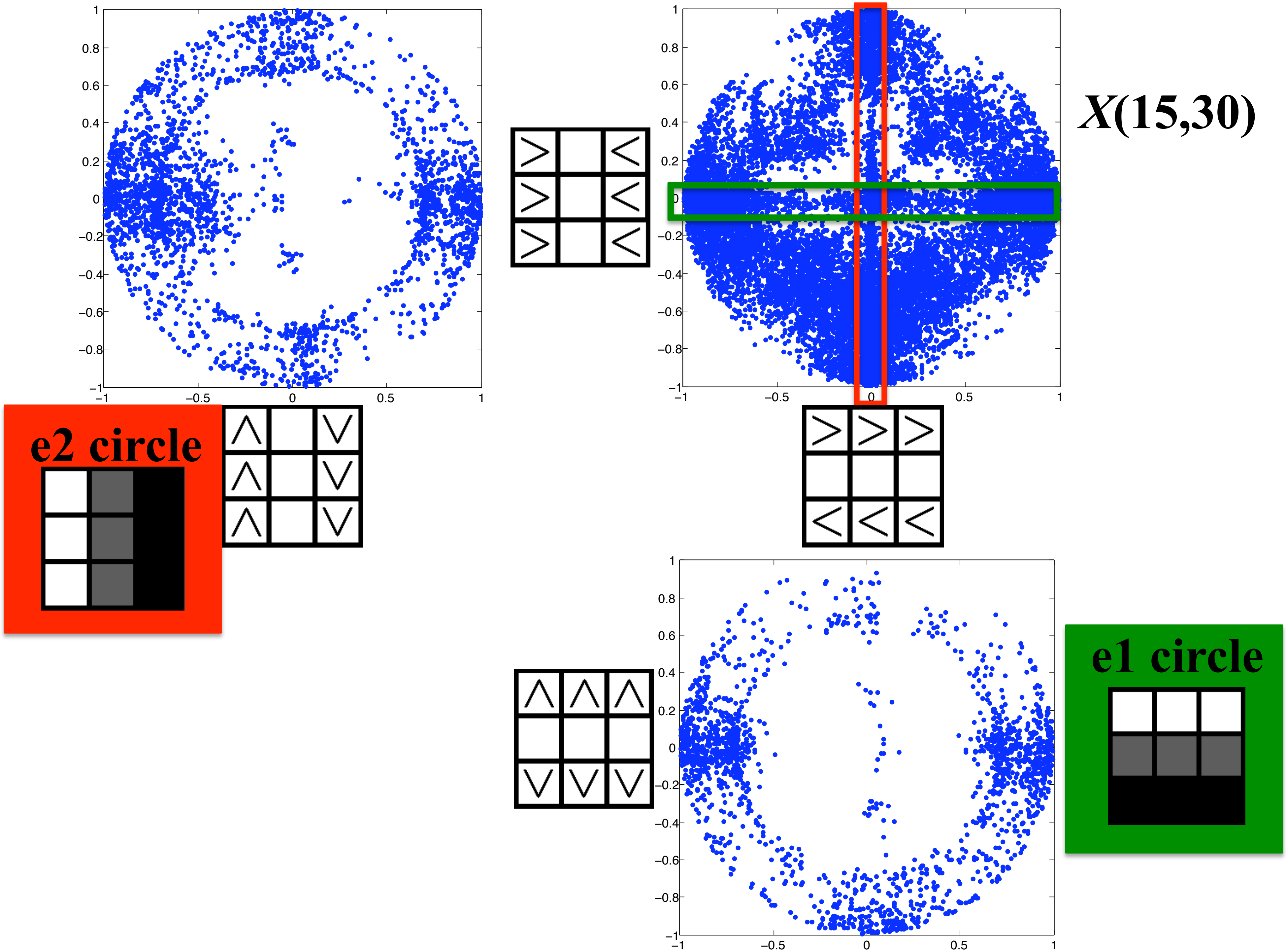
Range primary circle



Horizontal flow circle

X(15,30)

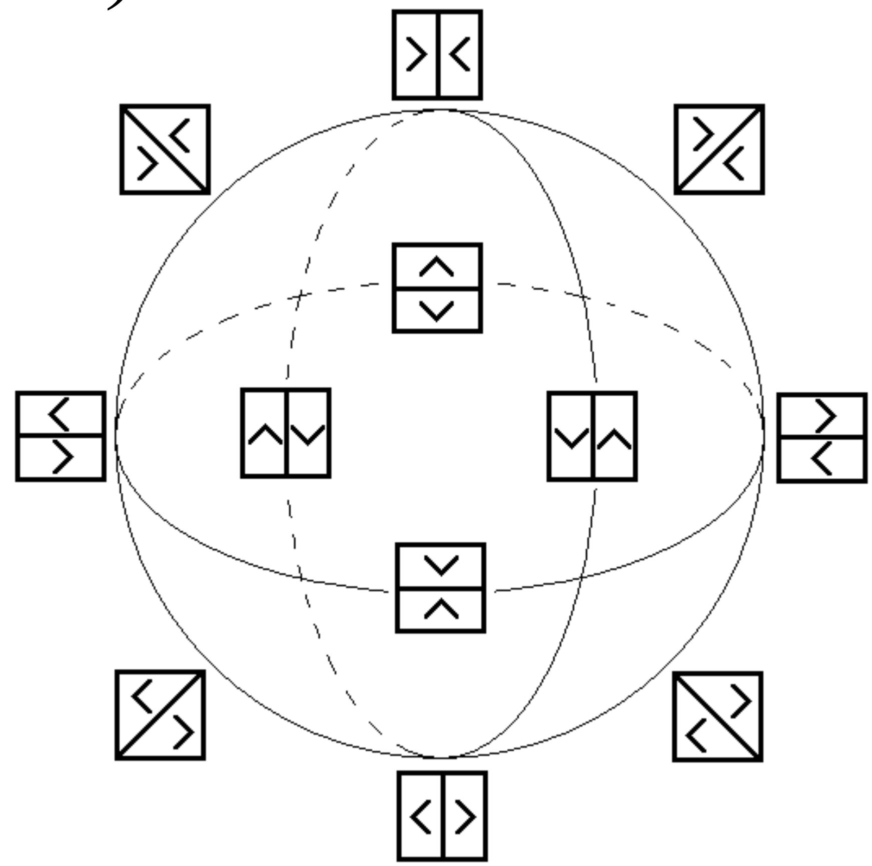




X(15,30)
e2 circle
e1 circle

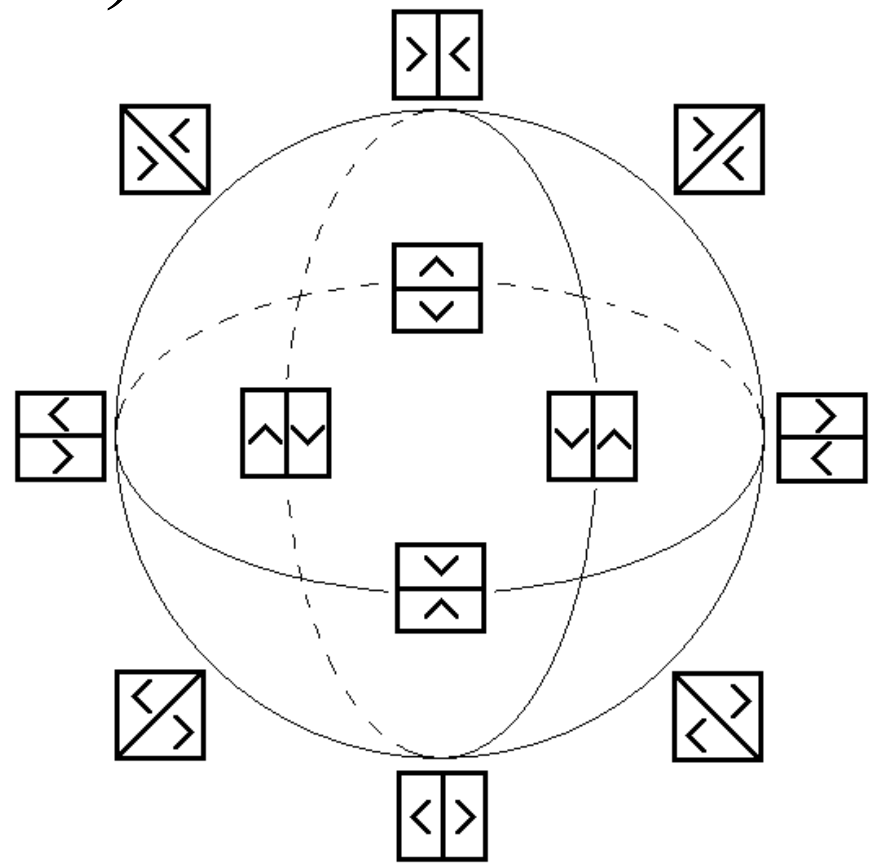
X(15,30)

- This 3-circle model is not analogous to that for photographs.
- Two parameters: range patch α in S^1 and camera translation β in S^1.



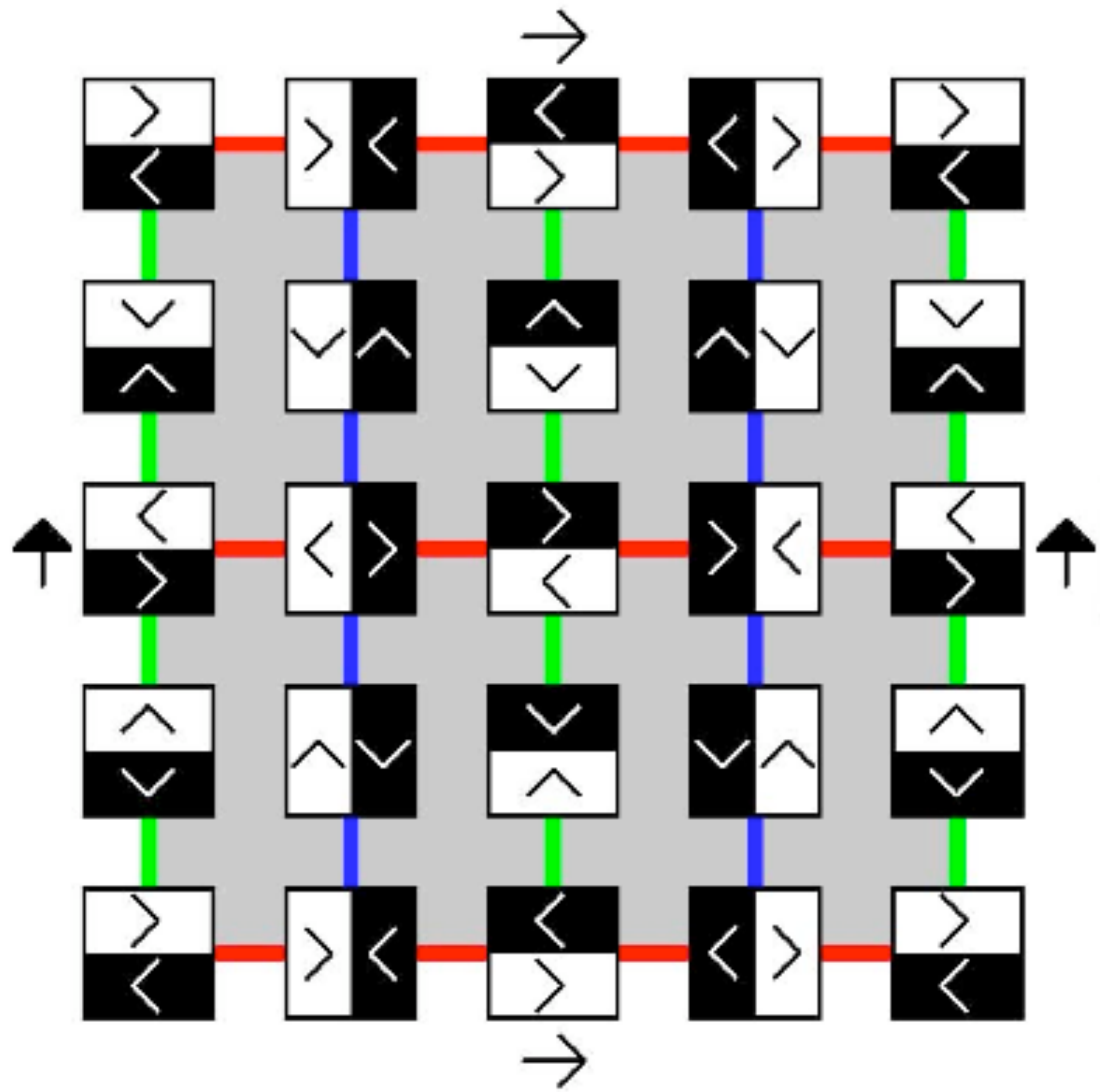
X(15,30)

- Fixing most common β's and varying α over S^1 gives horizontal flow circle.
- Fixing most common α's and varying β over S^1 gives e1 and e2 circles.



- What space do we get if we vary both α and β simultaneously?

Flow torus

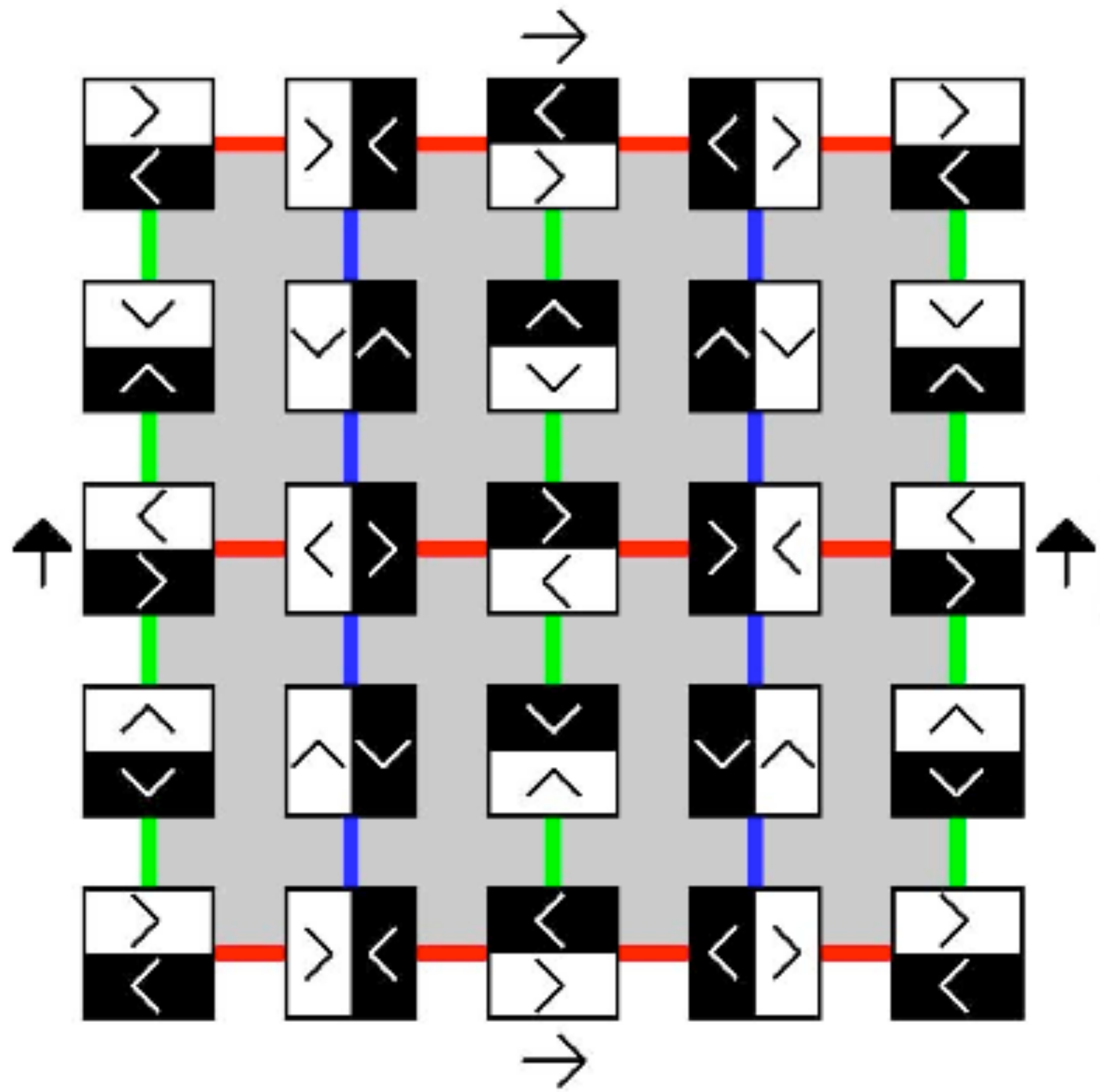


Camera translation β in S^1

Range patch α in S^1

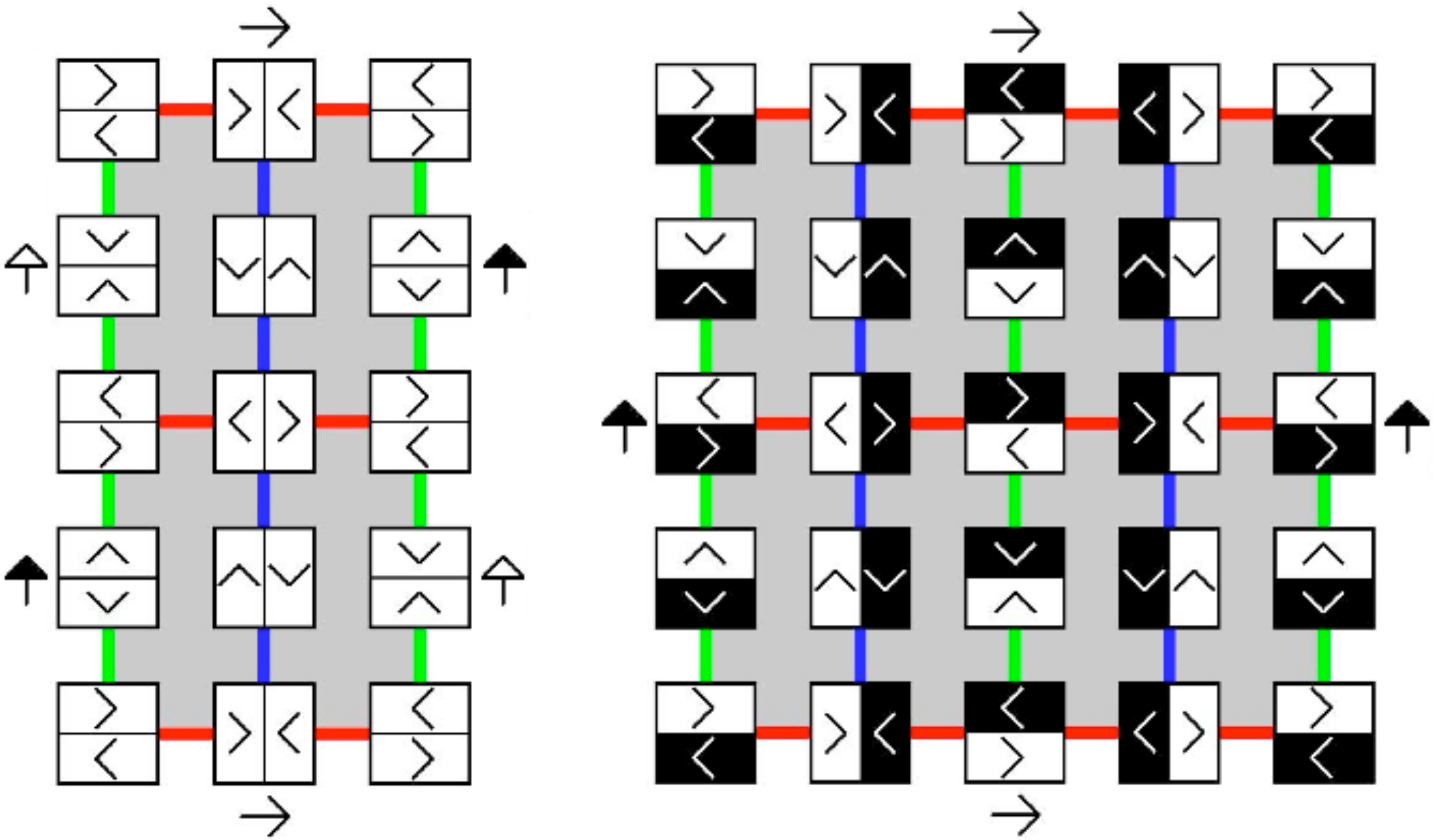
Flow torus

- Each flow patch appears twice.
  - Horizontal flow circle red.
  - e1 circle green.
  - e2 circle blue.



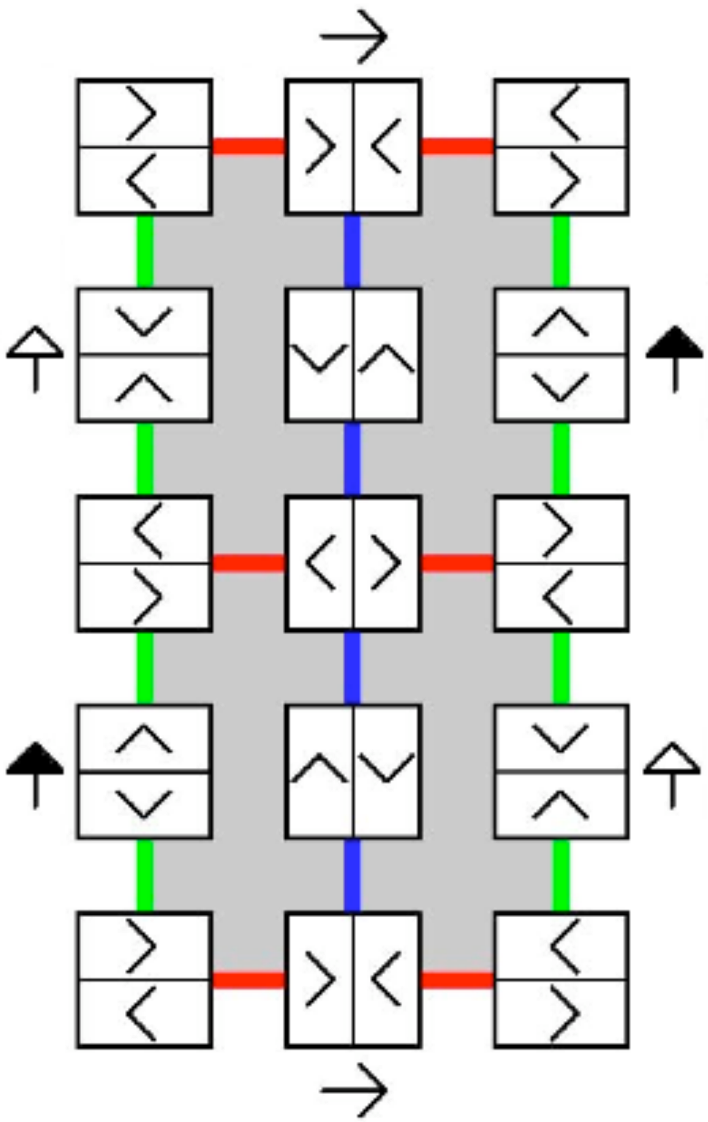
Identify antipodal points…

Flow torus

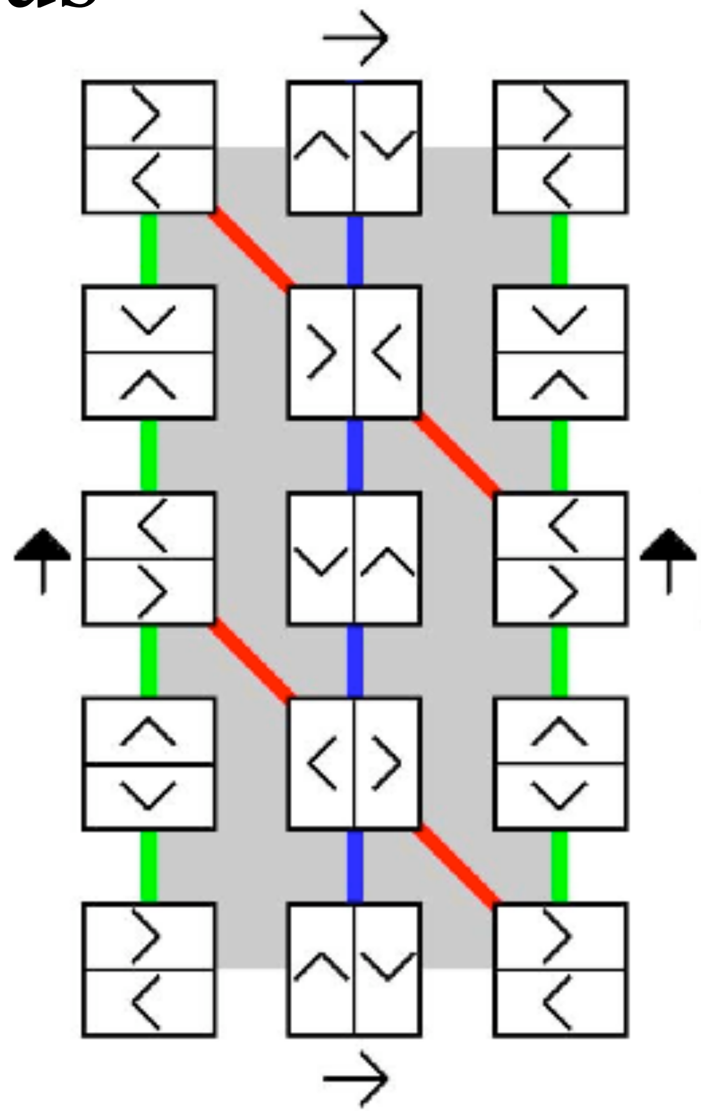


Identify antipodal points…

Flow torus

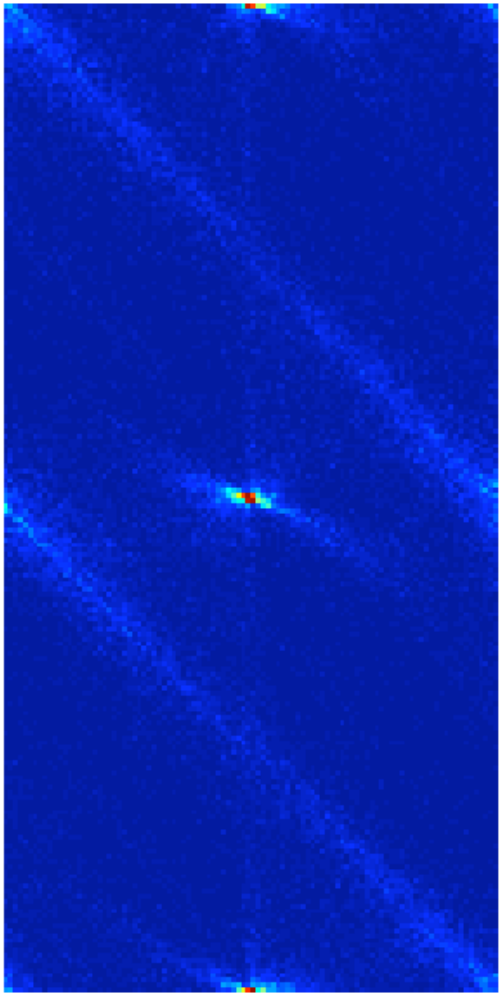


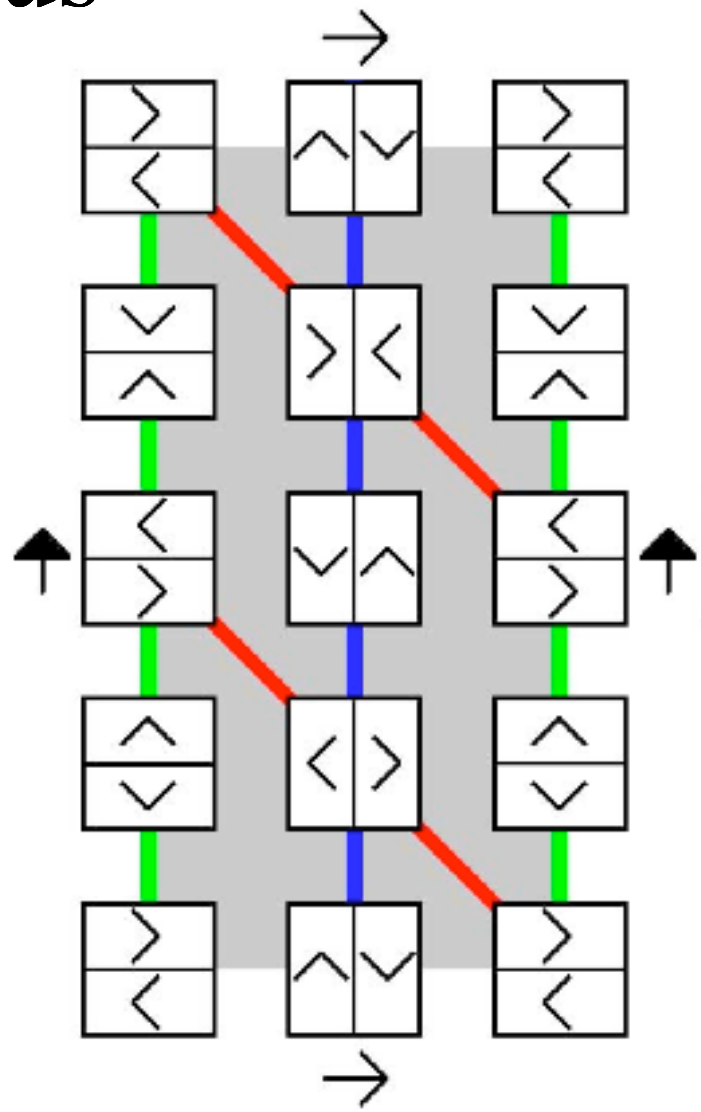
… drag right edge downwards…



get a torus.

Flow torus





- Project & plot heat map of all 50,000 high-contrast flow patches (not just a core subset).

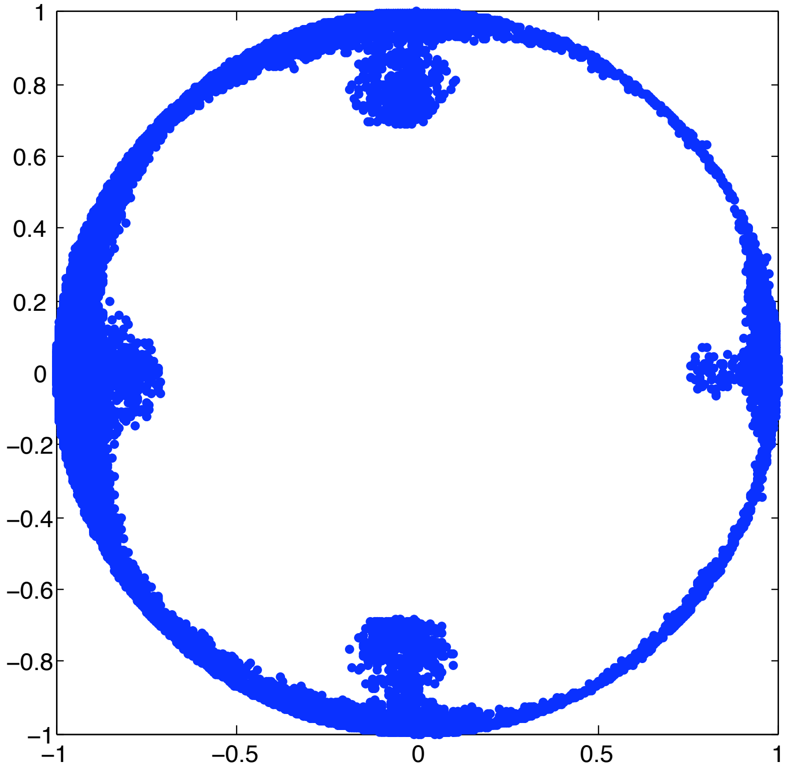
Why no torus barcodes?

- Answer 1: they're coming.
  - dataset with more vertical camera motion.
  - dataset with scene motion.
  - lower density cut k

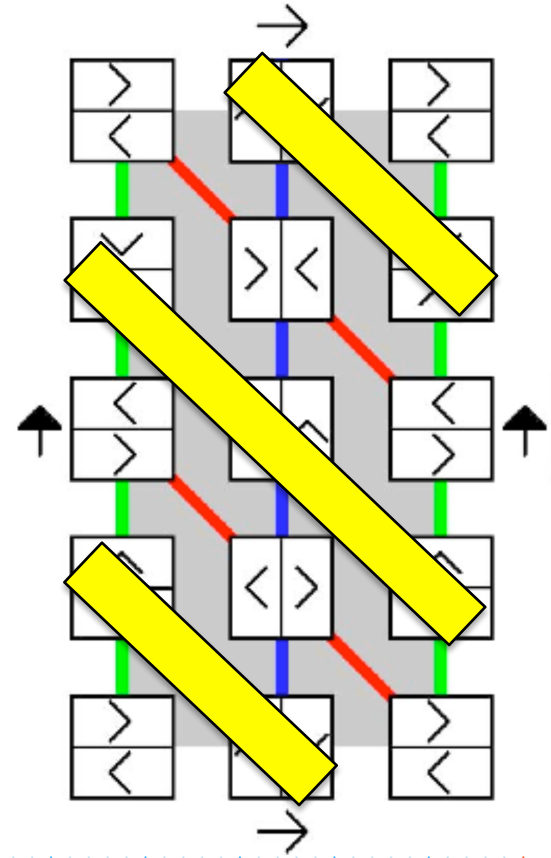
    (week to get $\boldsymbol{\rho}_k$ for million points)
  - more refined core subsets.

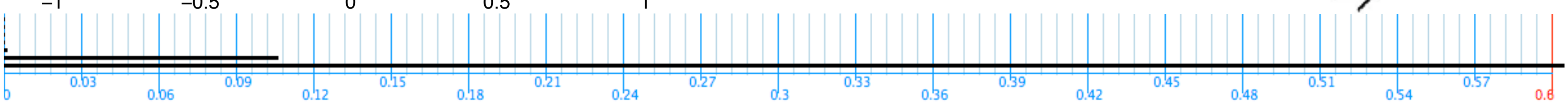
- Answer 2: don't need them.
  - used 1-skeleton to find torus model.
  - is torus model helpful?

Torus evidence: directional components

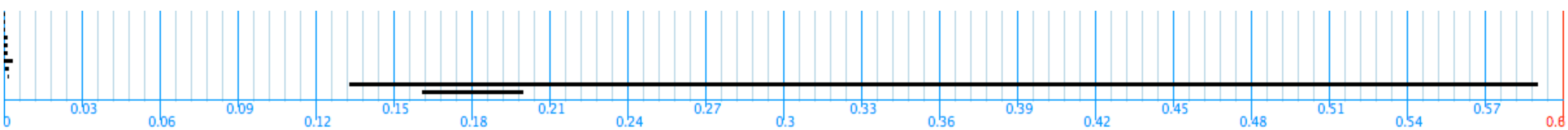


Find analog of horizontal flow circle for each directional component.





Betti plot: Dimension 0

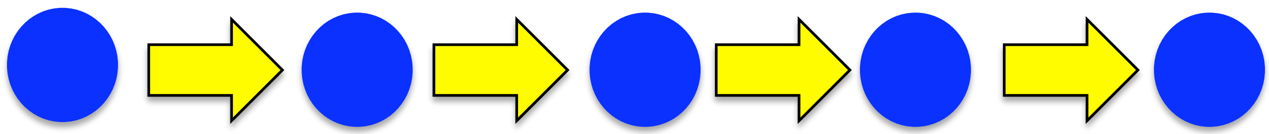


Betti plot: Dimension 1

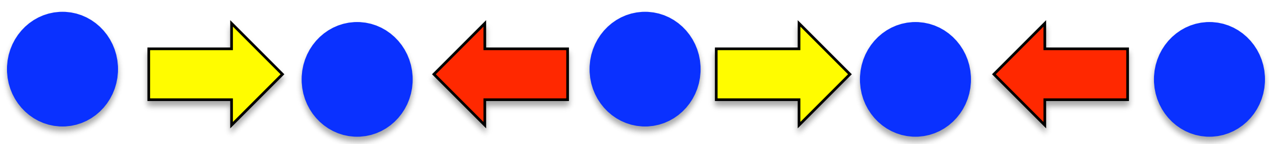
Possible applications

- Indoor vs outdoor indentification
  - different textures on flow torus or circles
  - use Fourier coefficients to distinguish?
- Estimation
  - incorporate torus model to get improved optical flow prior (Bayes' Theorem)

Zigzag persistence example



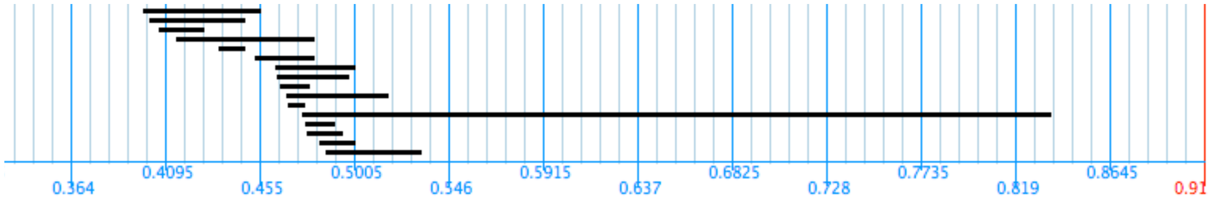
Persistence module of vector spaces and linear maps



Zigzag module

- Any map can point forwards or backwards.
- One of many uses: consistency between trials.

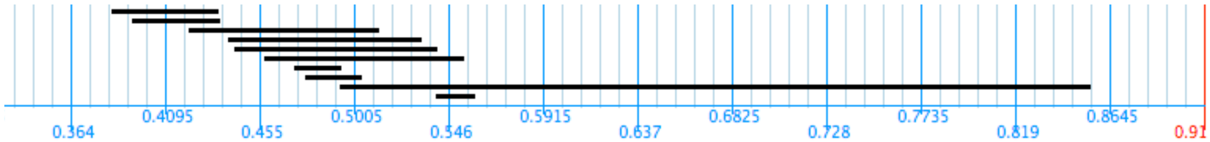
$LW_0(L_1,t)$



Betti plot L1: Dimension 1

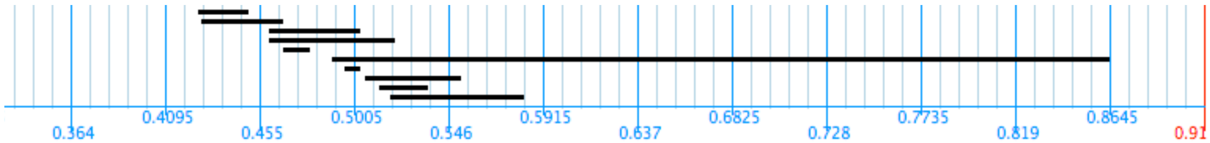
- Recovering the same circle each time or three different circles?

$LW_0(L_2,t)$

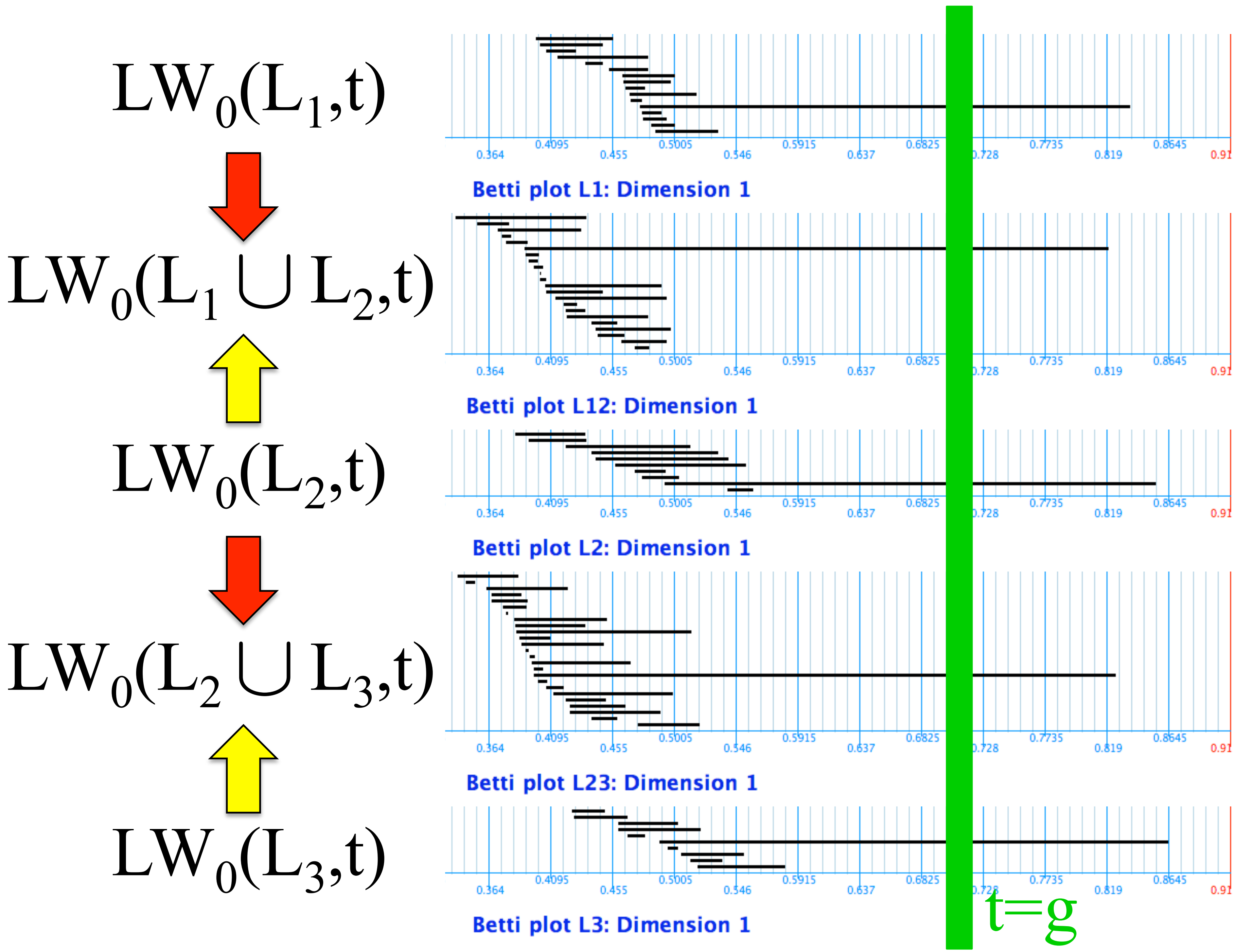


Betti plot L2: Dimension 1

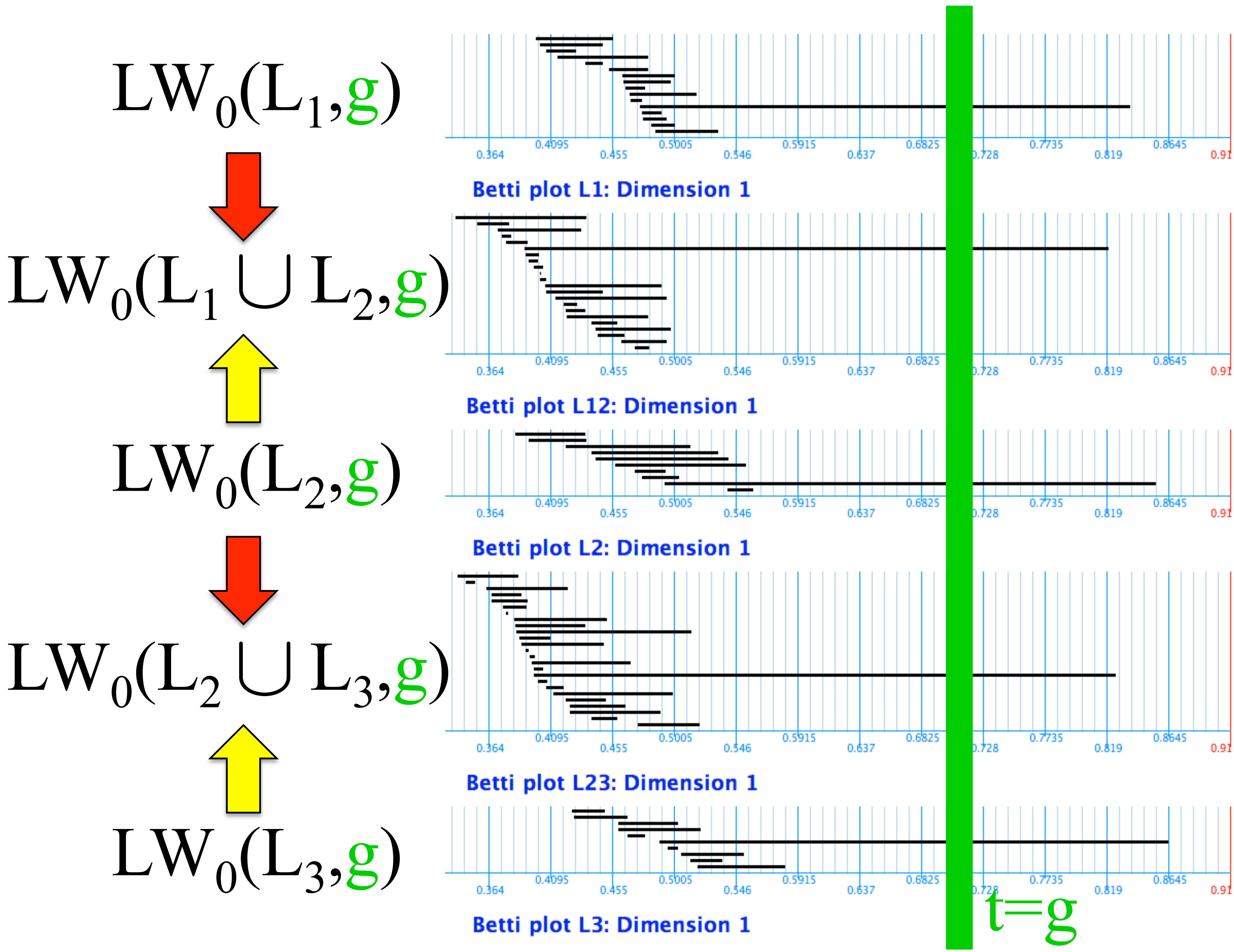
$LW_0(L_3,t)$



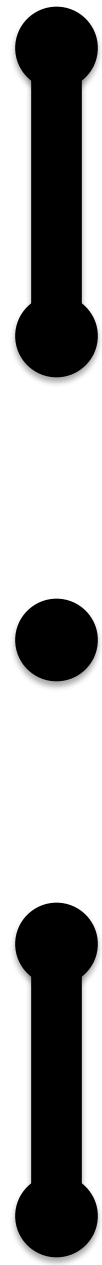
Betti plot L3: Dimension 1



LW0(L1,t)
LW0(L1 ∪ L2,t)
LW0(L2,t)
LW0(L2 ∪ L3,t)
LW0(L3,t)
Betti plot L1: Dimension 1
Betti plot L12: Dimension 1
Betti plot L2: Dimension 1
Betti plot L23: Dimension 1
Betti plot L3: Dimension 1
0.364
0.4095
0.455
0.5005
0.546
0.5915
0.637
0.6825
0.728
0.7735
0.819
0.8645
0.91
t=g


$LW_0(L_1, g)$
$LW_0(L_1 \cup L_2, g)$
$LW_0(L_2, g)$
$LW_0(L_2 \cup L_3, g)$
$LW_0(L_3, g)$
0.364
0.4095
0.455
0.5005
0.546
0.5915
0.637
0.6825
0.728
0.7735
0.819
0.8645
Betti plot L1: Dimension 1
Betti plot L12: Dimension 1
Betti plot L2: Dimension 1
Betti plot L23: Dimension 1
Betti plot L3: Dimension 1
t=g

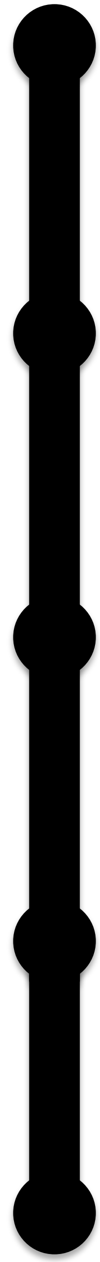
$$
\begin{tikzcd}
LW_0(L_1,g) \arrow[d] \\
LW_0(L_1 \cup L_2,g) \\
LW_0(L_2,g) \arrow[u] \arrow[d] \\
LW_0(L_2 \cup L_3,g) \\
LW_0(L_3,g) \arrow[u]
\end{tikzcd}
$$



Zigzag barcode for H_1

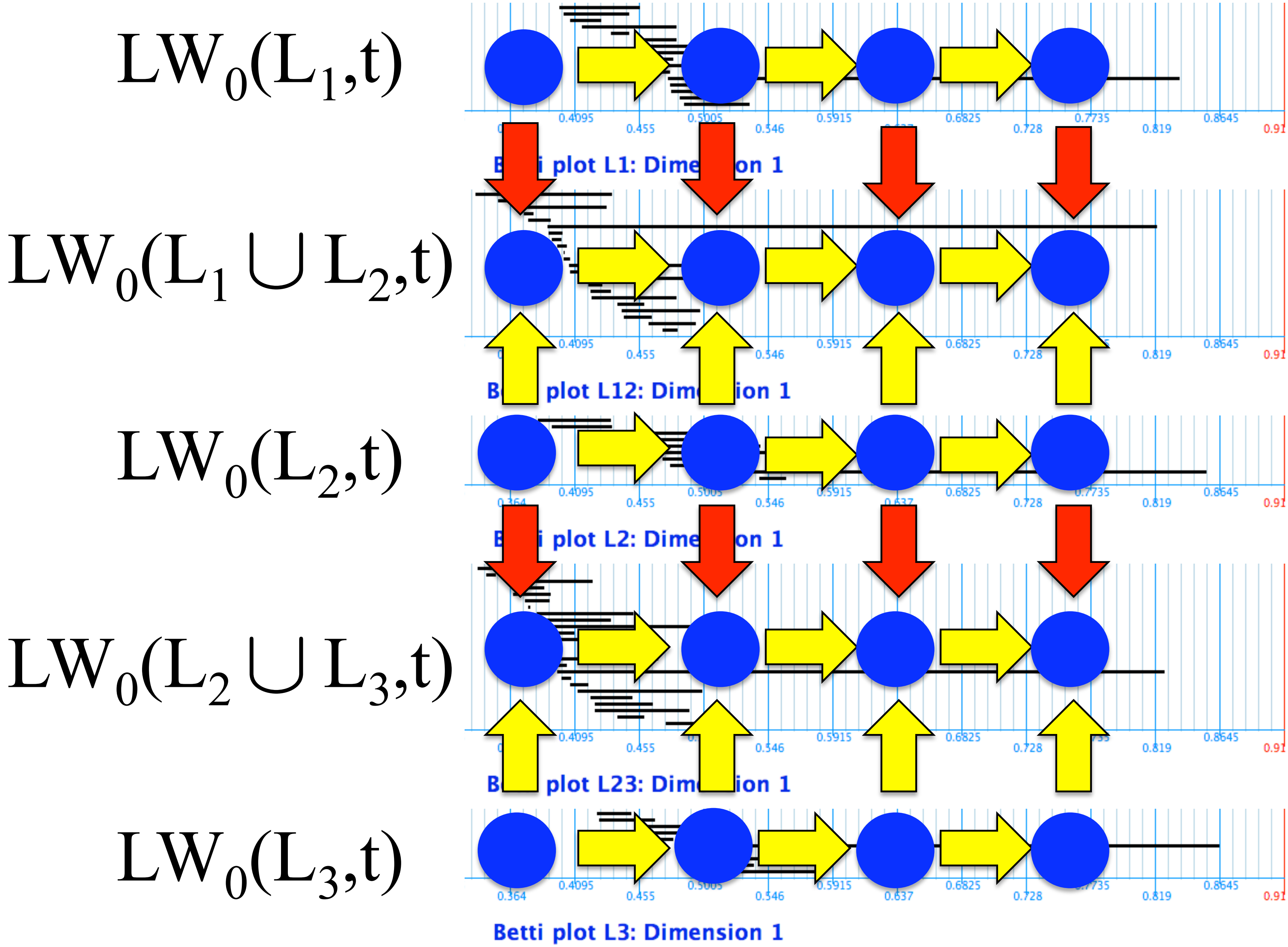
- NOT what we get.
- WOULD be 3 circles.
- Lack of consistency.

$$
\begin{tikzcd}
LW_0(L_1, g) \arrow[d] \\
LW_0(L_1 \cup L_2, g) \\
LW_0(L_2, g) \arrow[u] \arrow[d] \\
LW_0(L_2 \cup L_3, g) \\
LW_0(L_3, g) \arrow[u]
\end{tikzcd}
$$



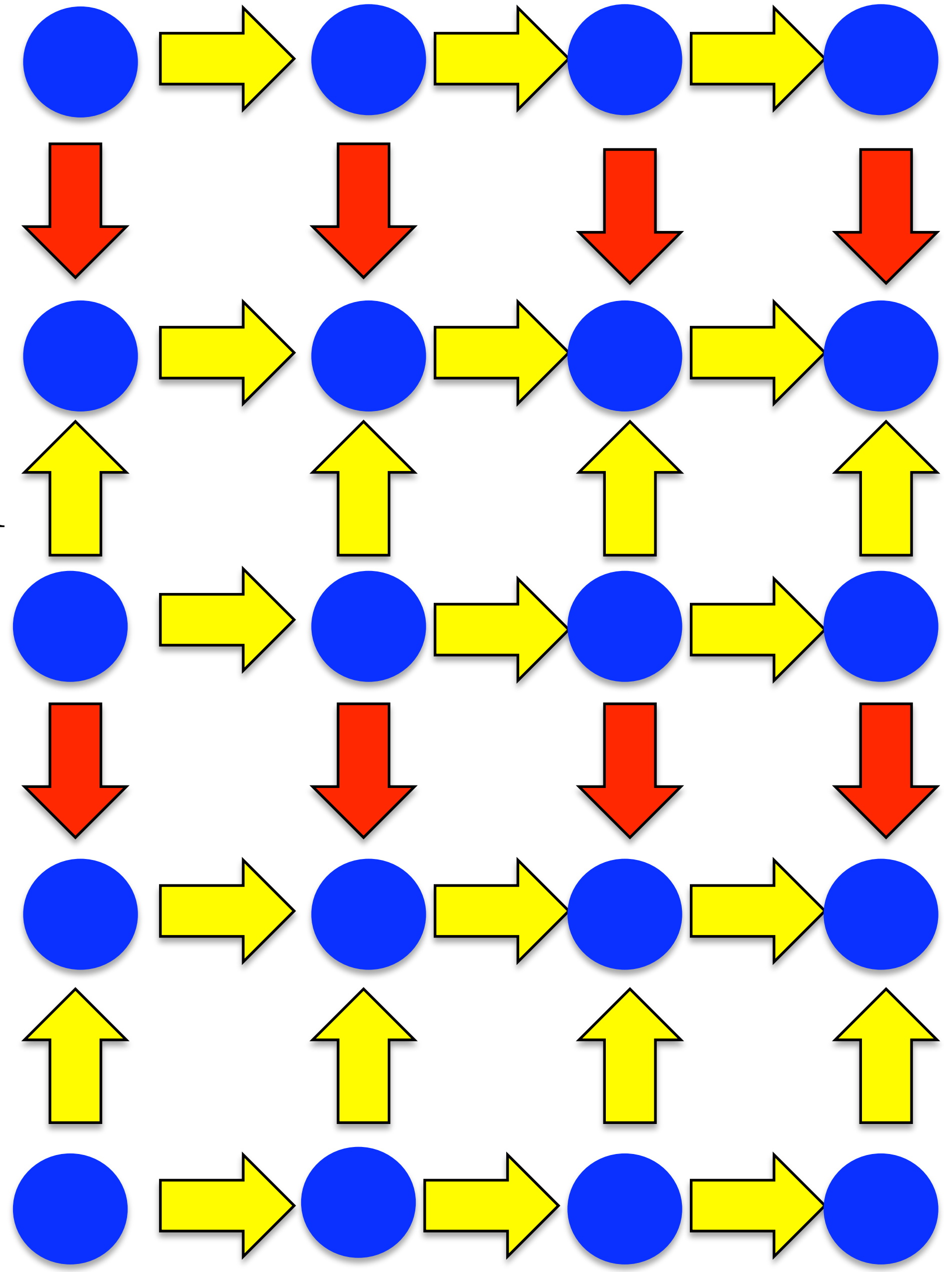
Zigzag barcode for H_1

- We get one circle.
- Consistency.



$LW_0(L_1,t)$
$LW_0(L_1 \cup L_2,t)$
$LW_0(L_2,t)$
$LW_0(L_2 \cup L_3,t)$
$LW_0(L_3,t)$
Betti plot L3: Dimension 1

Multidimensional zigzag module



Announcement: JPlex discussion (too real?)

- Morse filtrations
- Preprocessing m-files
- BeanShell
- Computation speed
- Zigzag implementation and O(n^3) vs. O(nm^2)
- Increasing Java heap size
- Source code
- Improving software and tutorial

- Tuesday 3:45 ?