# Operations in Multi-Dimensional Neural Fields

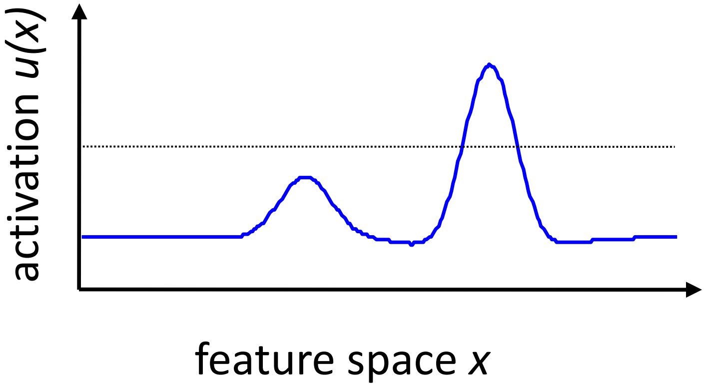
Summer School Neural Dynamics for Cognitive Robotics

August 2014

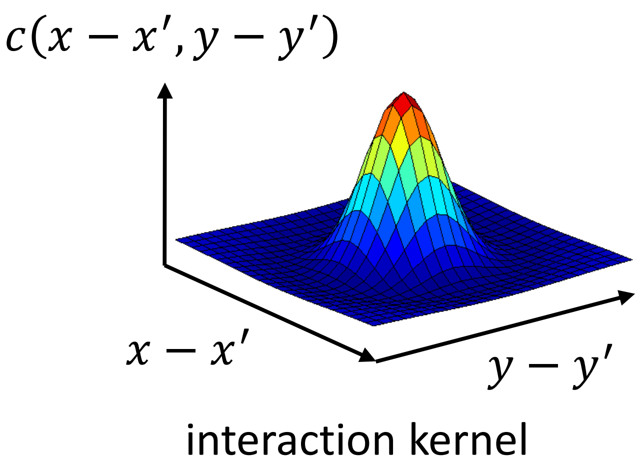

Sebastian Schneegans

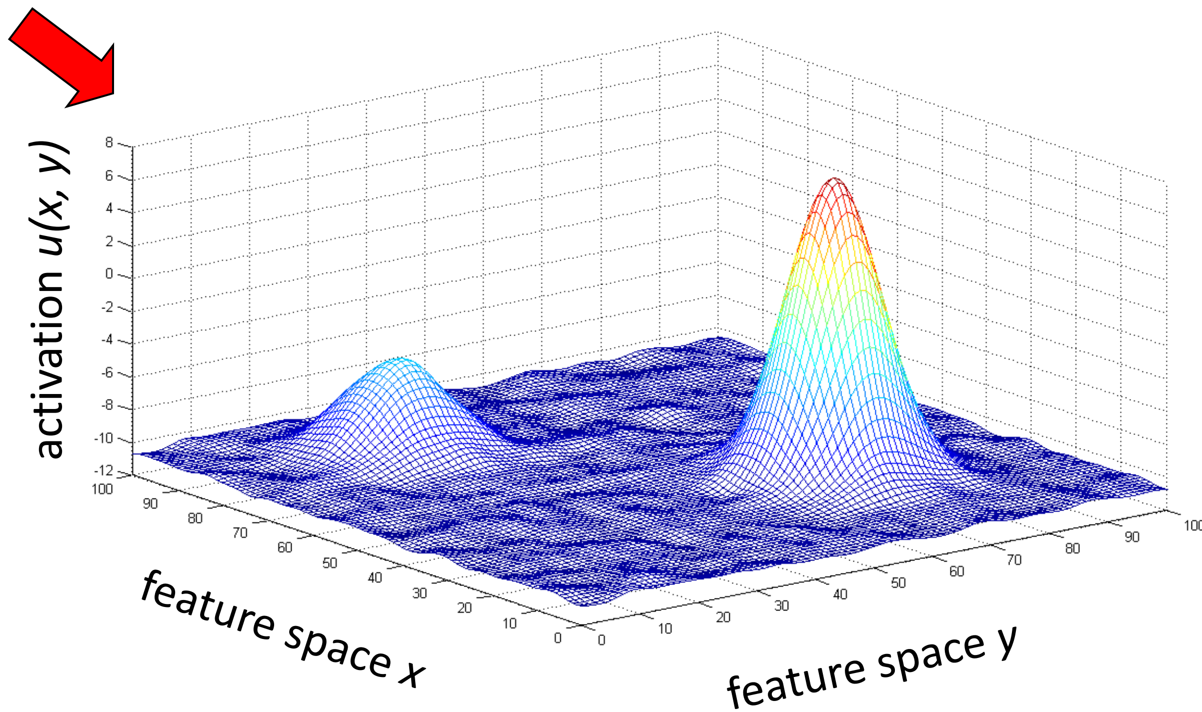
sebastian.schneegans@ini.rub.de

# Multi-dimensional fields




$$\tau\dot{u}(x,y) = -u(x,y) + h + s(x,y) + \iint c(x-x', y-y')\, g\big(u(x',y')\big)\, dx'dy'$$



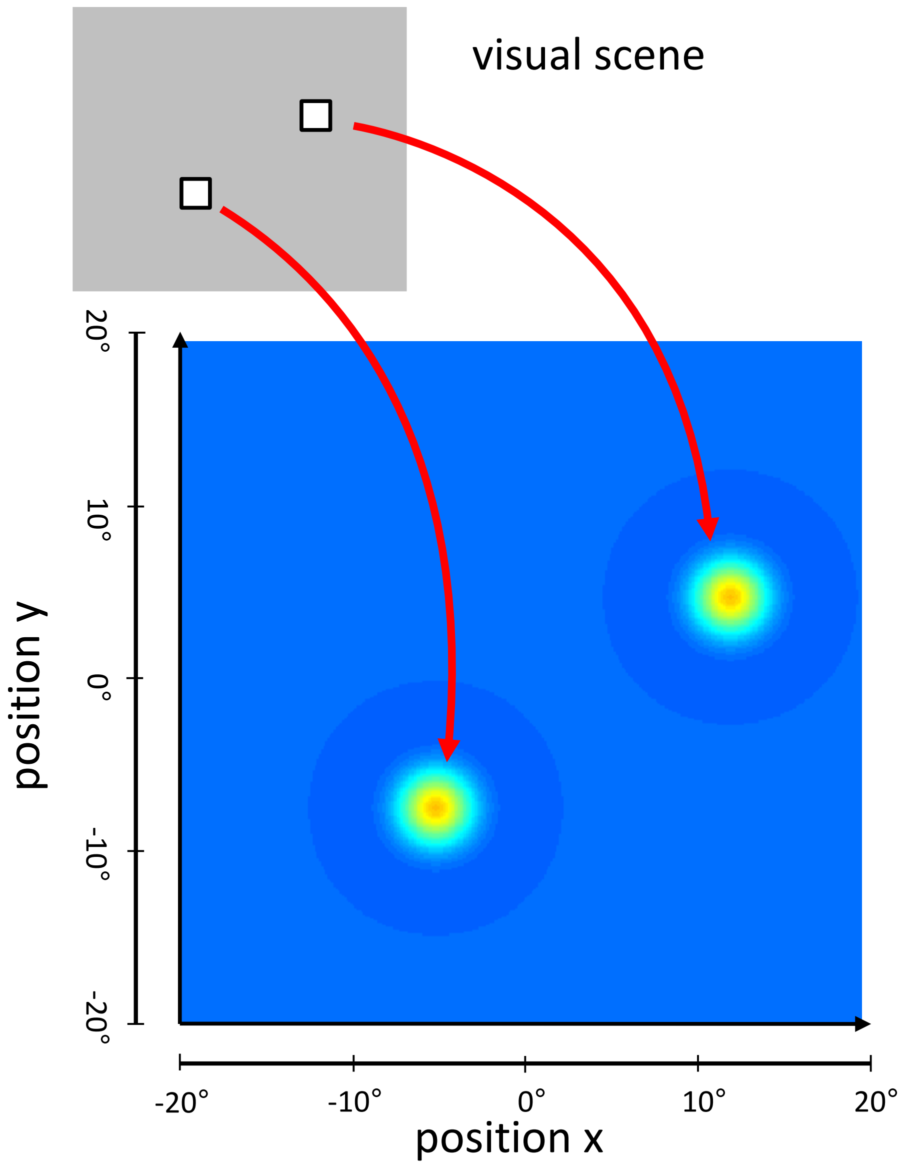



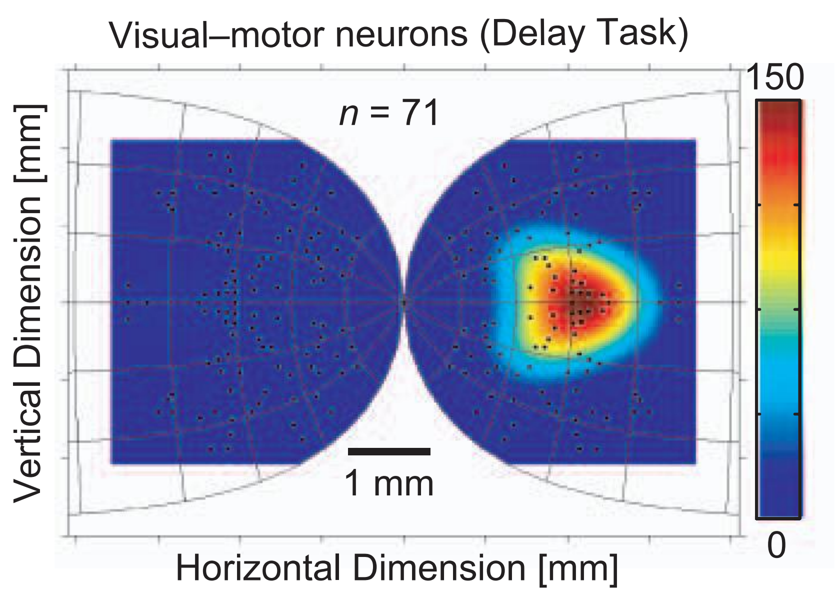

- extension to multi-dimensional feature spaces mathematically straightforward
- requires interaction kernel of the same dimensionality

# Multi-dimensional fields

- multi-dimensional fields retain the same qualitative instabilities and stable states as one-dimensional fields
- specifically:
  - activation peaks stabilized by interactions
  - detection and selection decision
  - self-sustained peaks as working memory

# Multi-dimensional feature spaces



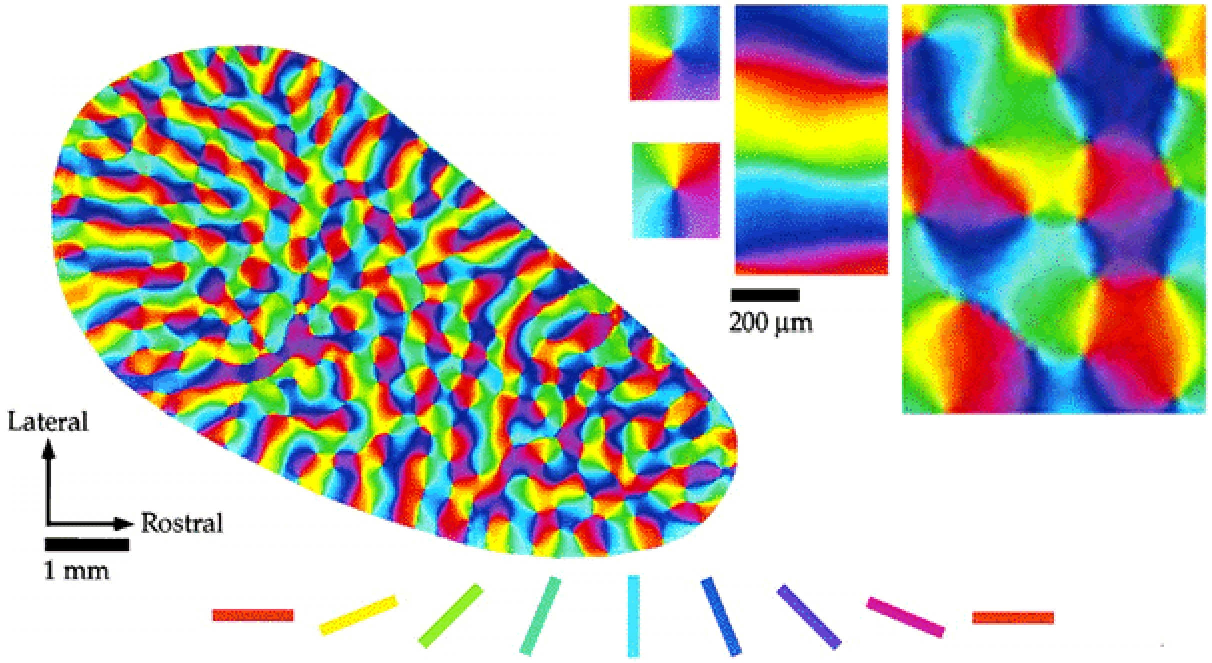




neural activity in superior colliculus
[Marino, Trappenberg, Dorris, Munoz 2012]

- some feature spaces are inherently multi-dimensional, simple example: visual space (2D)
- neural representations e.g. in superior colliculus (saccade planning)

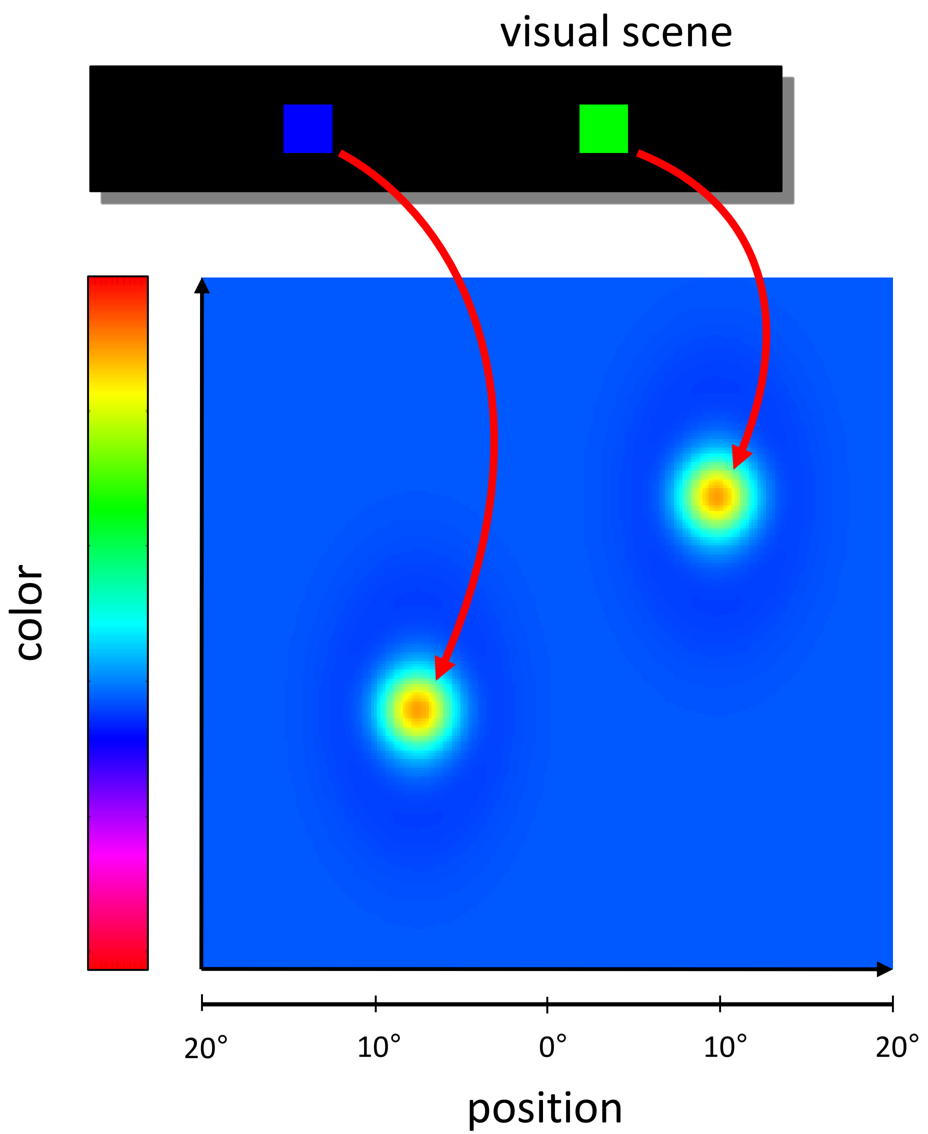
# Multi-dimensional feature spaces

- multi-dimensional feature spaces can also combine qualitatively different features
- example: early visual cortex, neurons with localized spatial receptive fields and sensitivity to surface features (orientation, spatial frequency, color, …)
- 3D representation (2D space + 1D surface feature) mapped onto 2D cortical surface



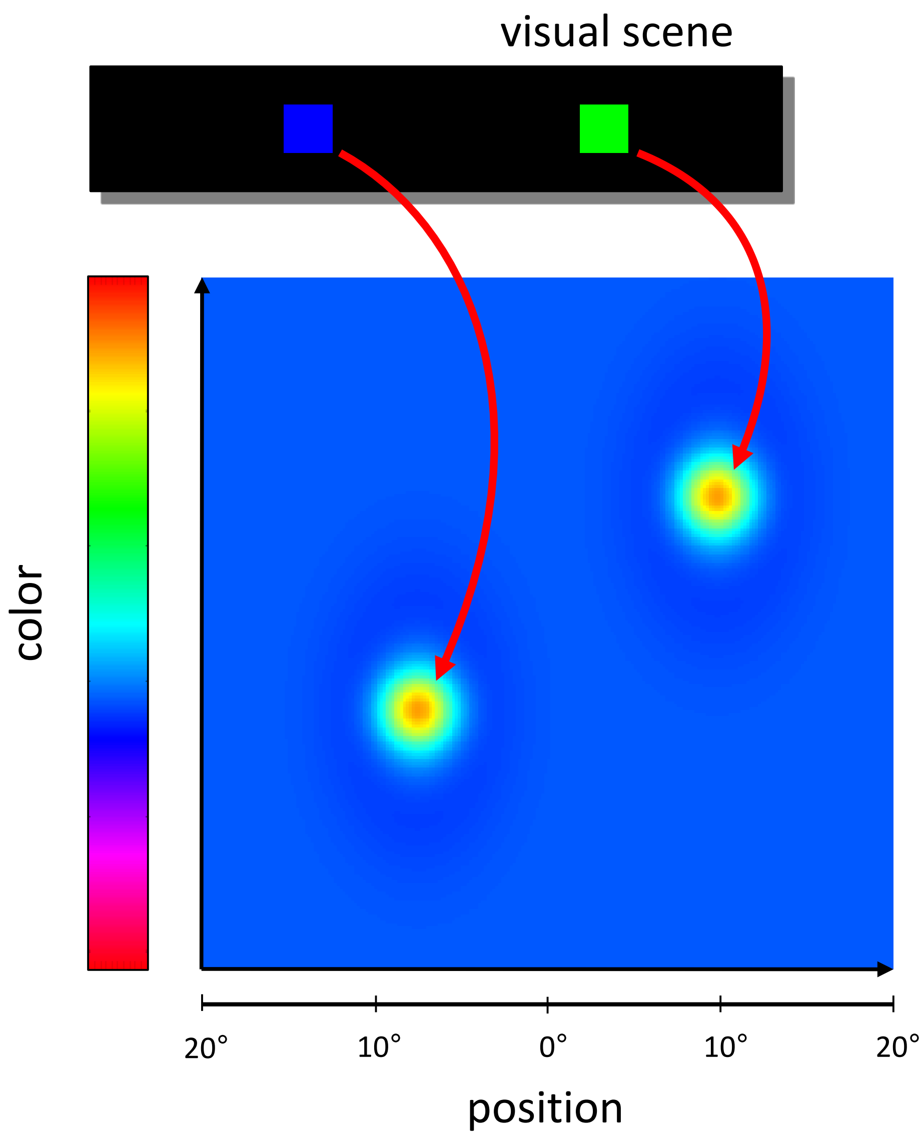

orientation map in tree shrew visual cortex [Alexander et al. 1999]

# Combining features in multi-dimensional fields



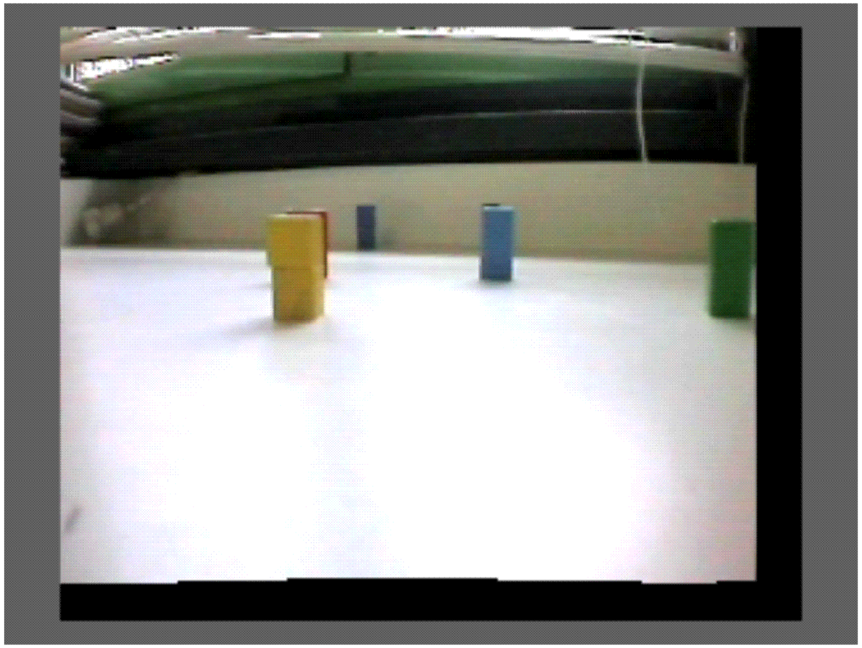

- neural fields defined over combinations of feature spaces
- requires to define metrics for lateral interactions in combined space
- fields ignore spatial arrangement of neurons on the cortical surface

# Combining features in multi-dimensional fields

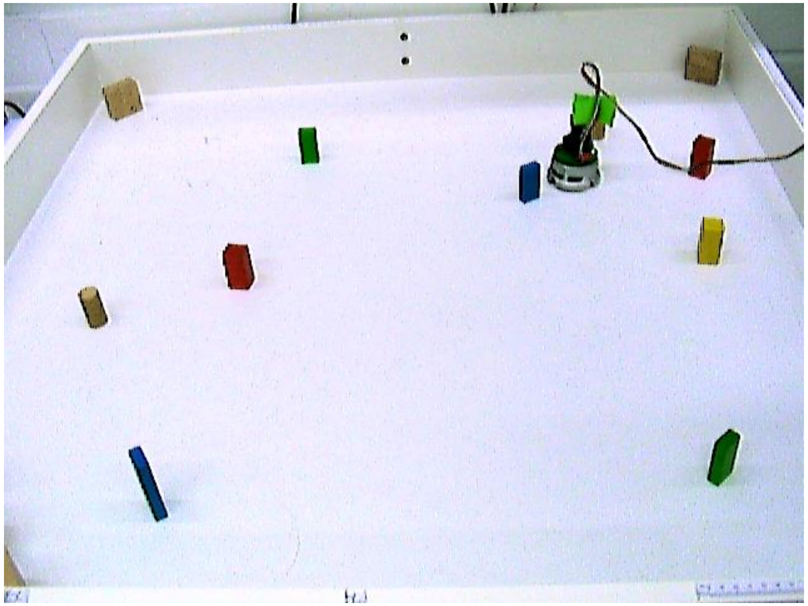



- visual stimuli can provide localized inputs
- ideally from feature detectors (like neurons in retina/visual cortex), in robotics often from algorithmic feature extraction

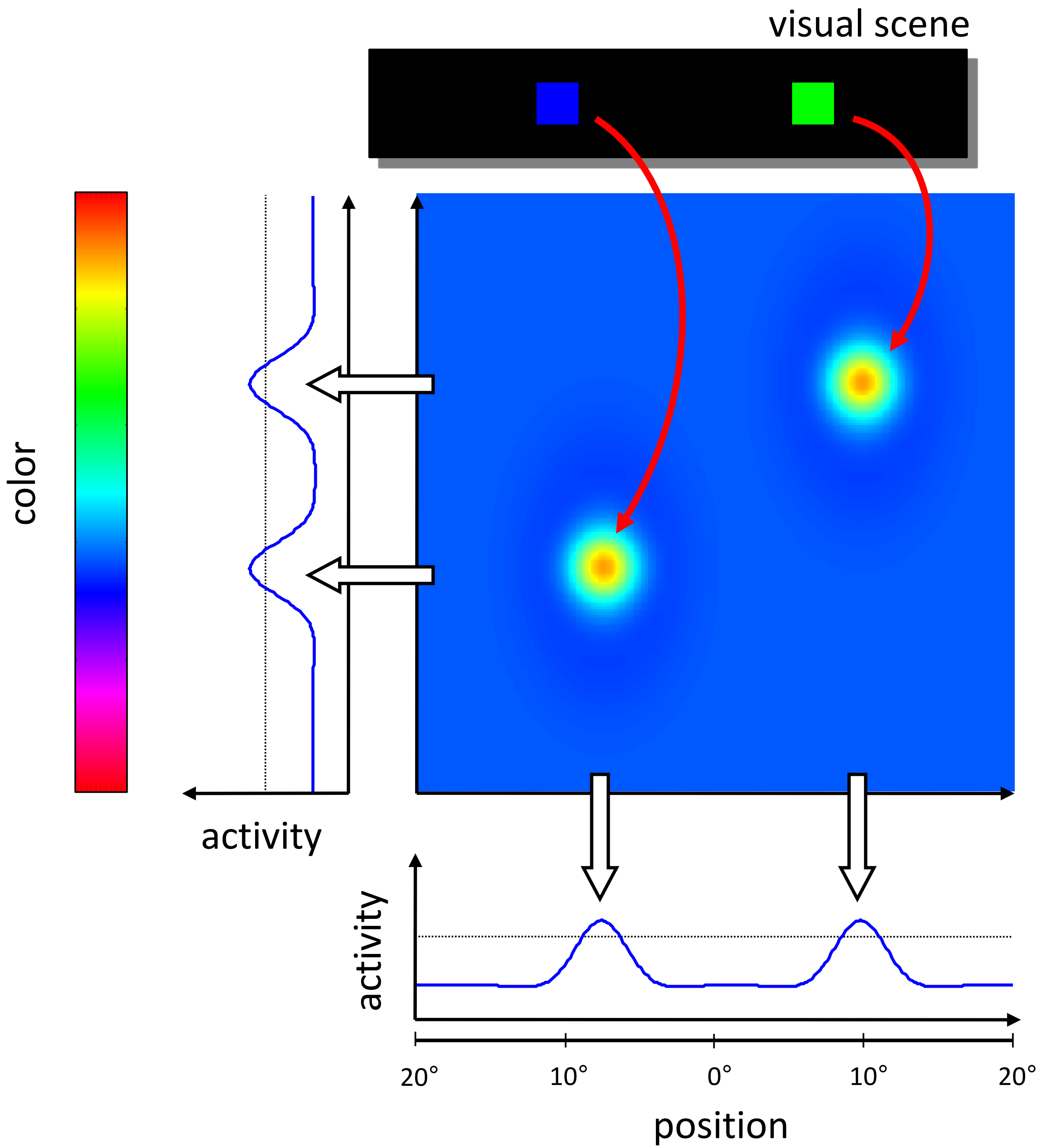
# 2D fields from camera input



- representation of full camera image/retinal input would require 3D field: two spatial and one color dimension
- can ignore one spatial dimension when only horizontal object position is relevant (e.g. for control of robot orientation on a planar surface)



# Reading out from 2D fields

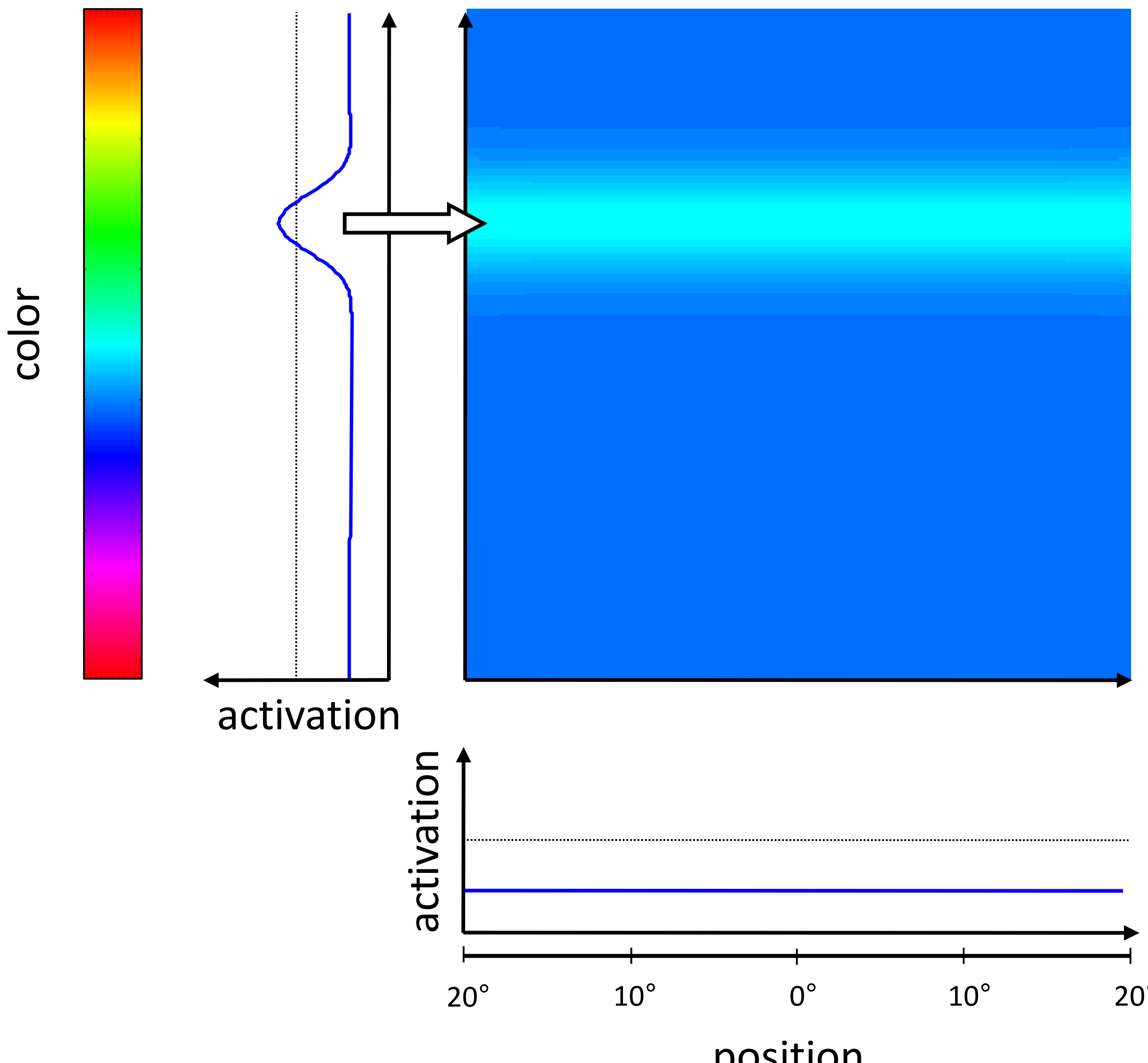



- 2D fields can interact with 1D fields
- first operation: read out of one feature dimension: integrate over discarded dimensions, e.g.

$$I_s(x) = \int f\big(u_v(x,y)\big)dy$$

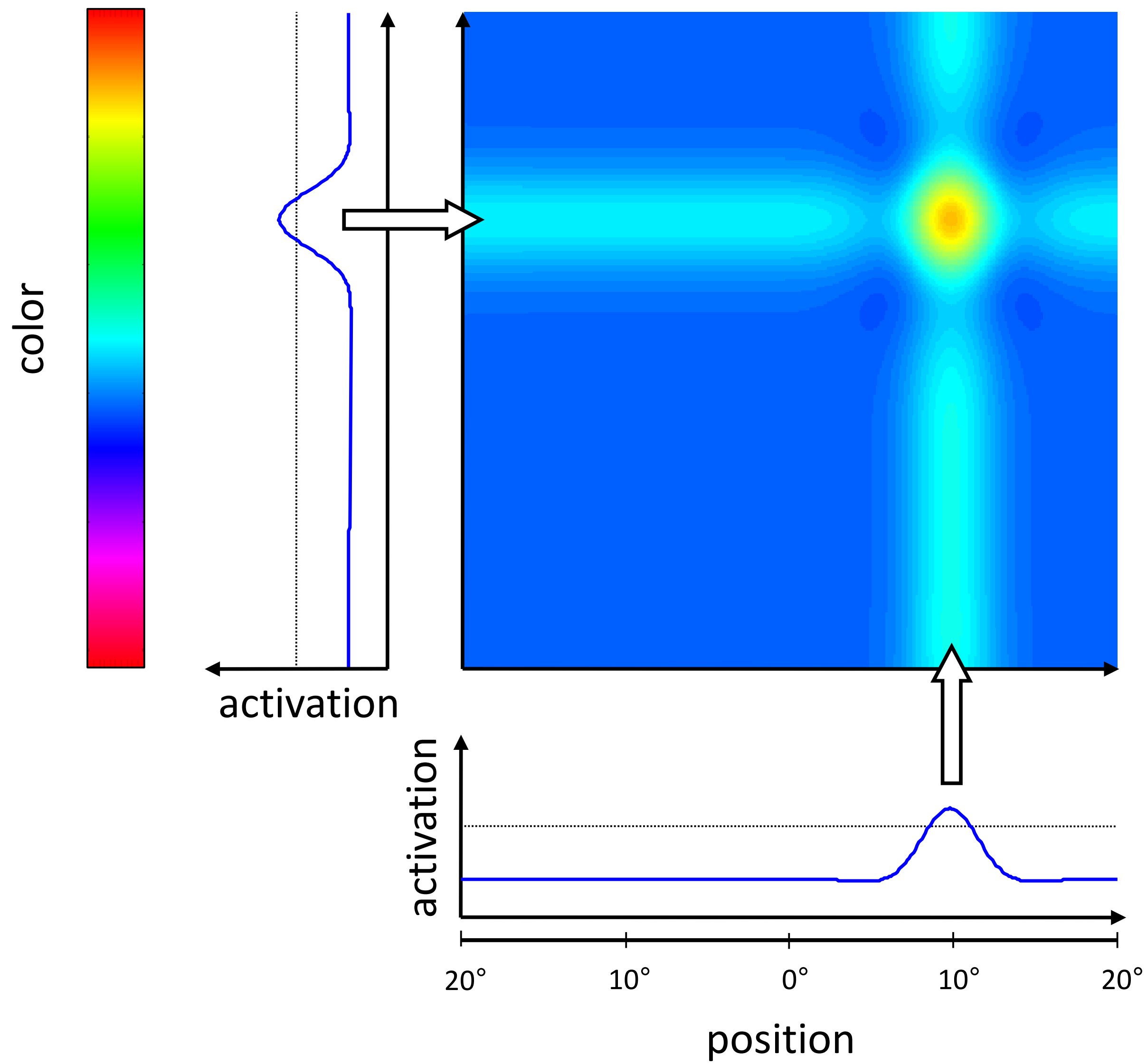
- often additional Gaussian convolution in projection for smoothness

# Projections to 2D fields




- projection from 1D to 2D: ridge input
- does not specify a location in the 2nd dimension, does not typically induce a peak (although field can be forced to form a peak)

# Projections to 2D fields




- intersections of ridges can induce a peak and produce a combined representation of multiple features

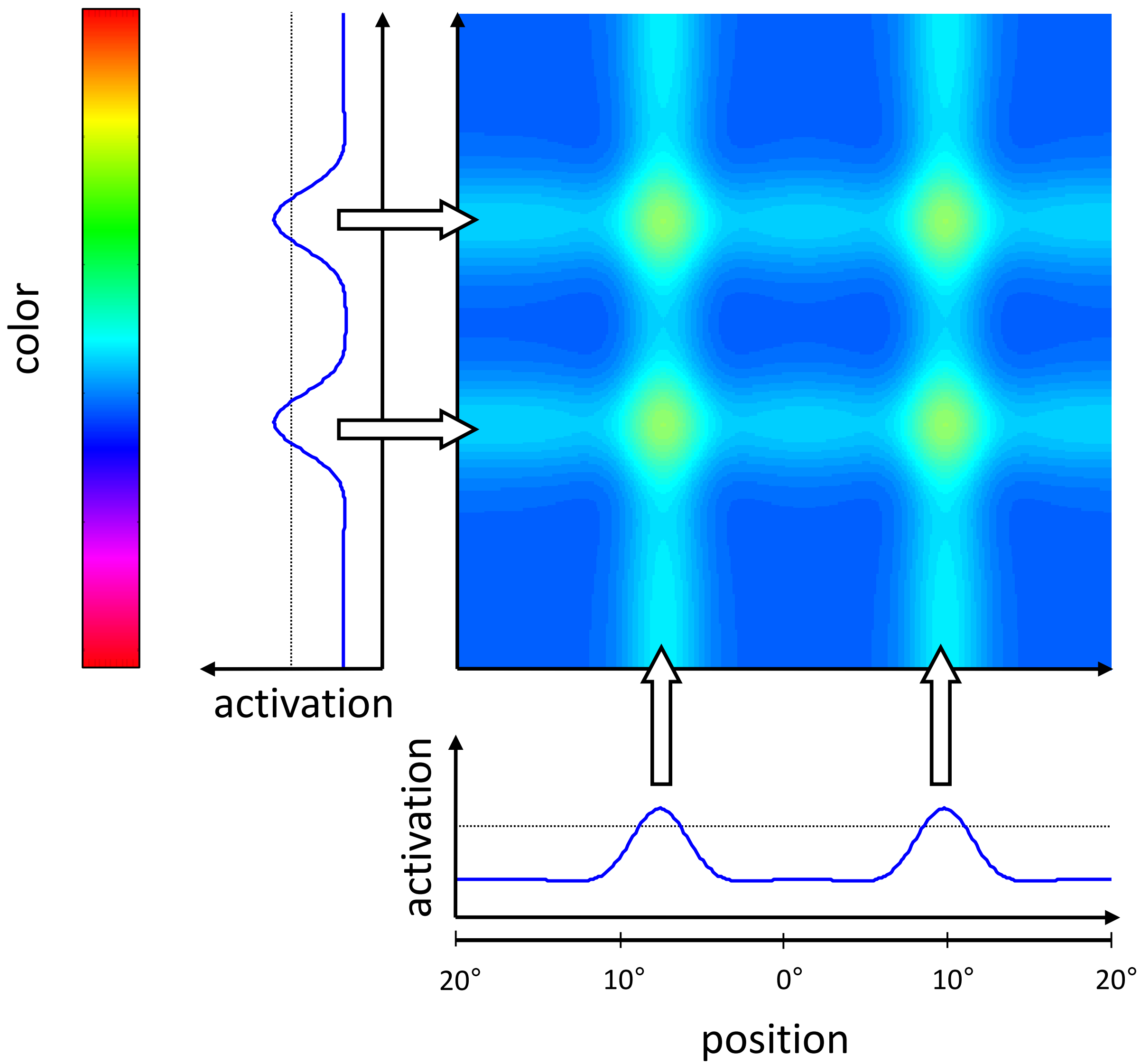
# Combined vs. separate representations

separate low-dimensional representations

- are much more compact (computationally less expensive / fewer neurons) – at sampling rate of 100 neurons per dimension, 200 neurons for two 1D fields, 10000 neurons for one 2D field)
- can represent individual feature values with the same precision/reliability as a 2D field

So why use 2D fields at all?

# Feature conjunctions

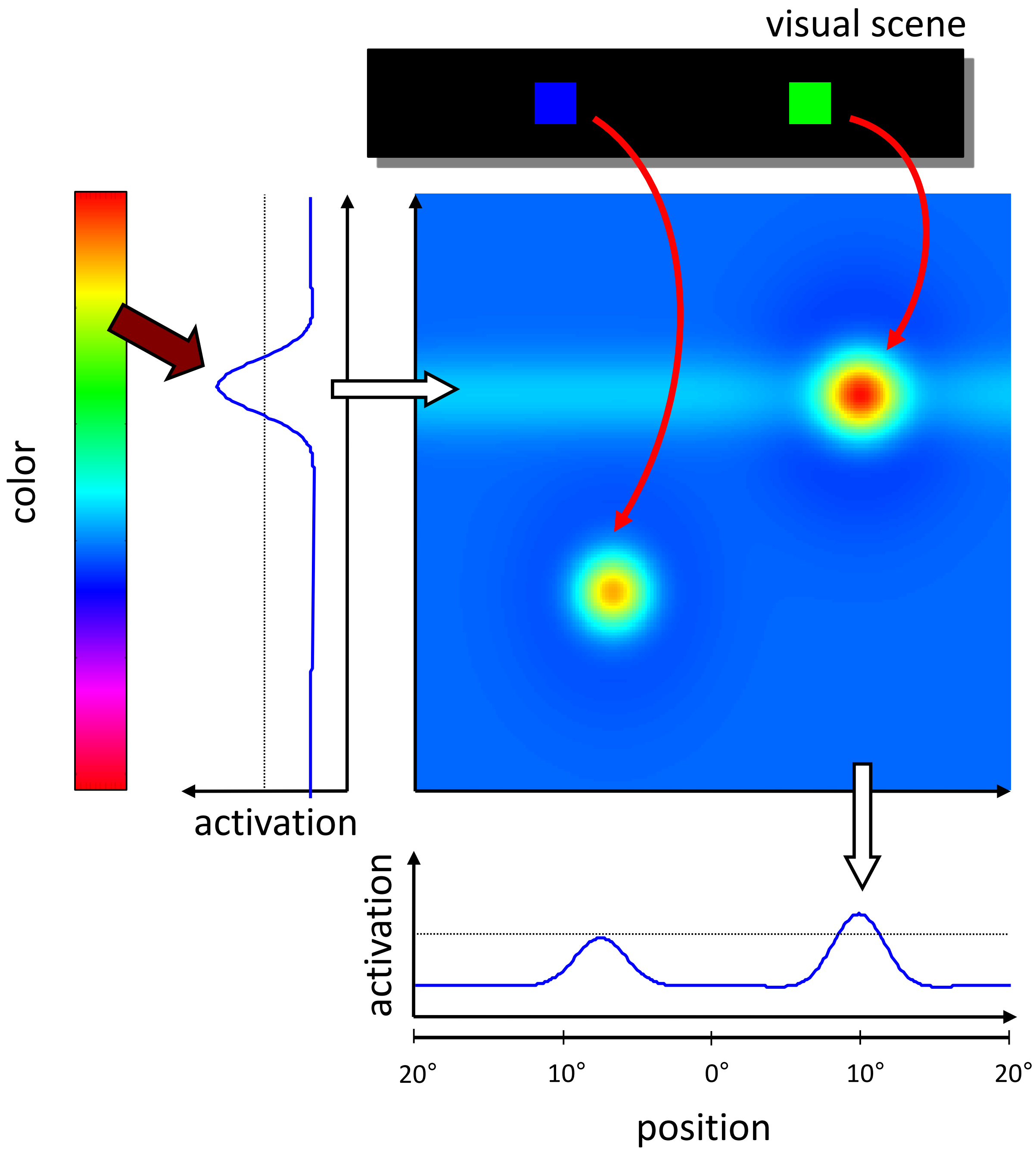



- low-dimensional representations do not capture feature conjunctions (binding problem)
- multiple ridge inputs can produce spurious peaks

# Combined vs. separate representations

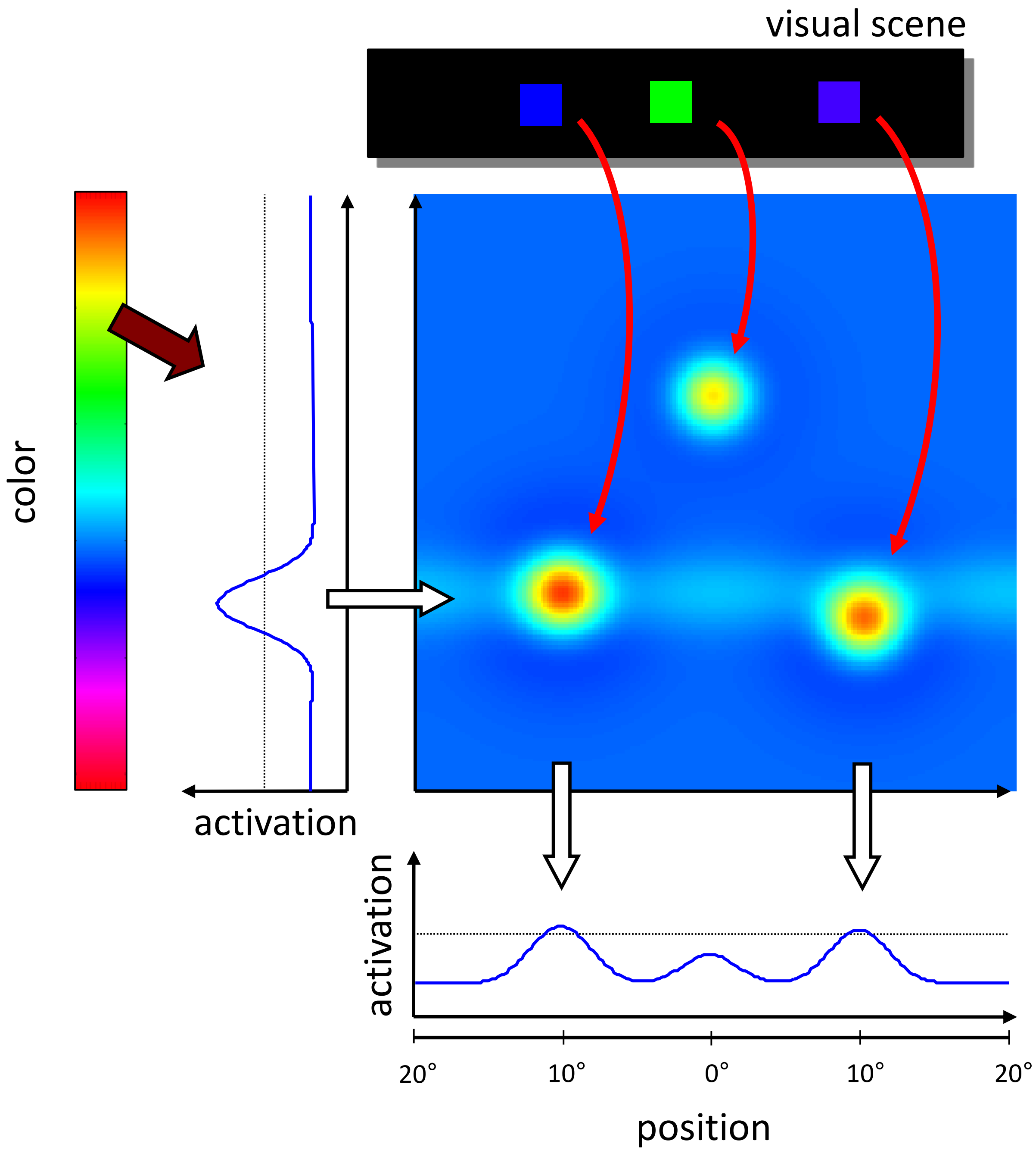
- multi-dimensional field are needed to represent feature conjunctions
- combining low-dimensional and high-dimensional fields can yield powerful architectures
- example: visual search, determine the location of an object with specified features

# Visual search



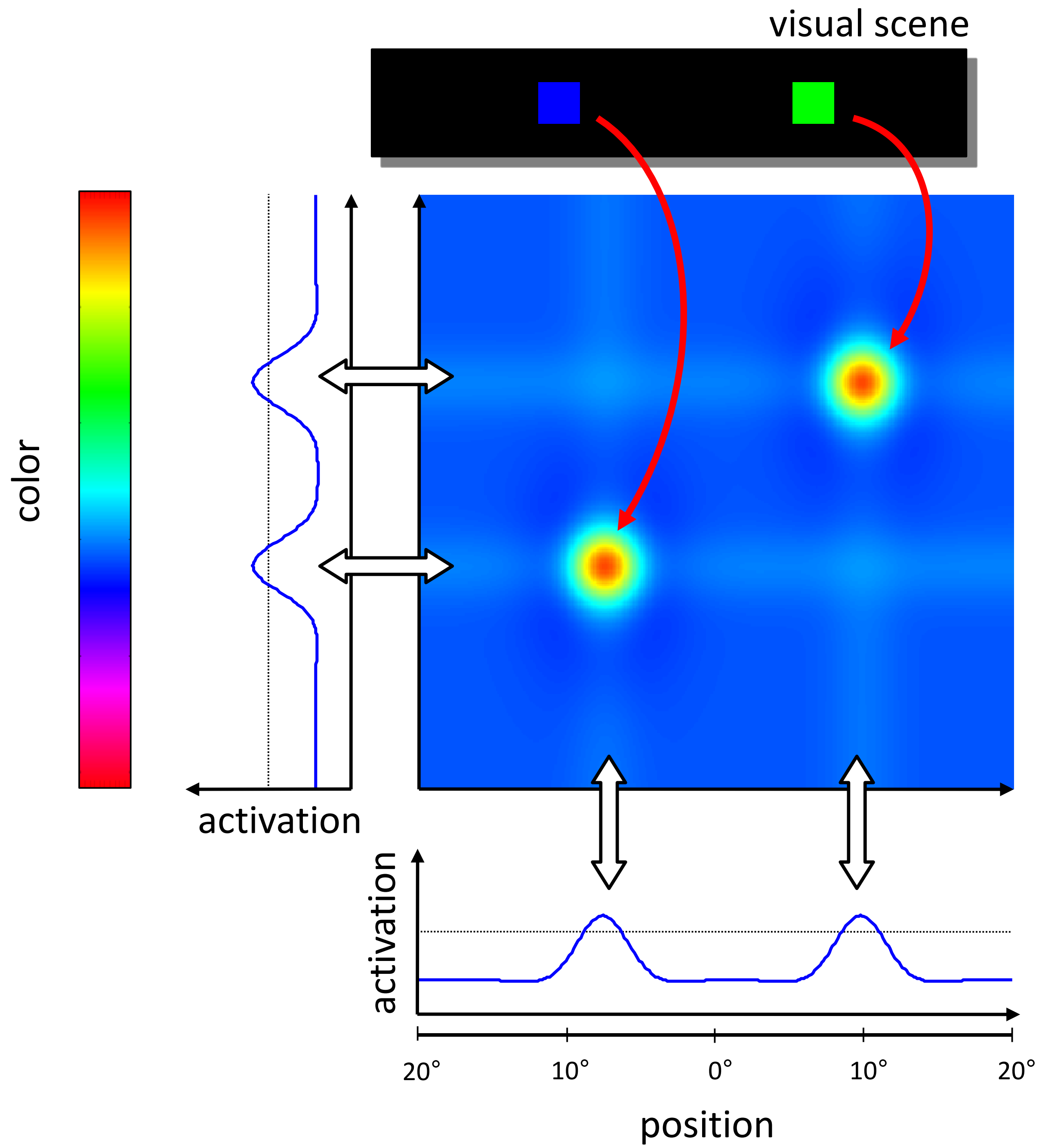

- if localized peaks are present in the 2D field, ridge input can be used to select one of them
- read-out along the 2nd dimension then allows to determine the associated feature

# Visual search



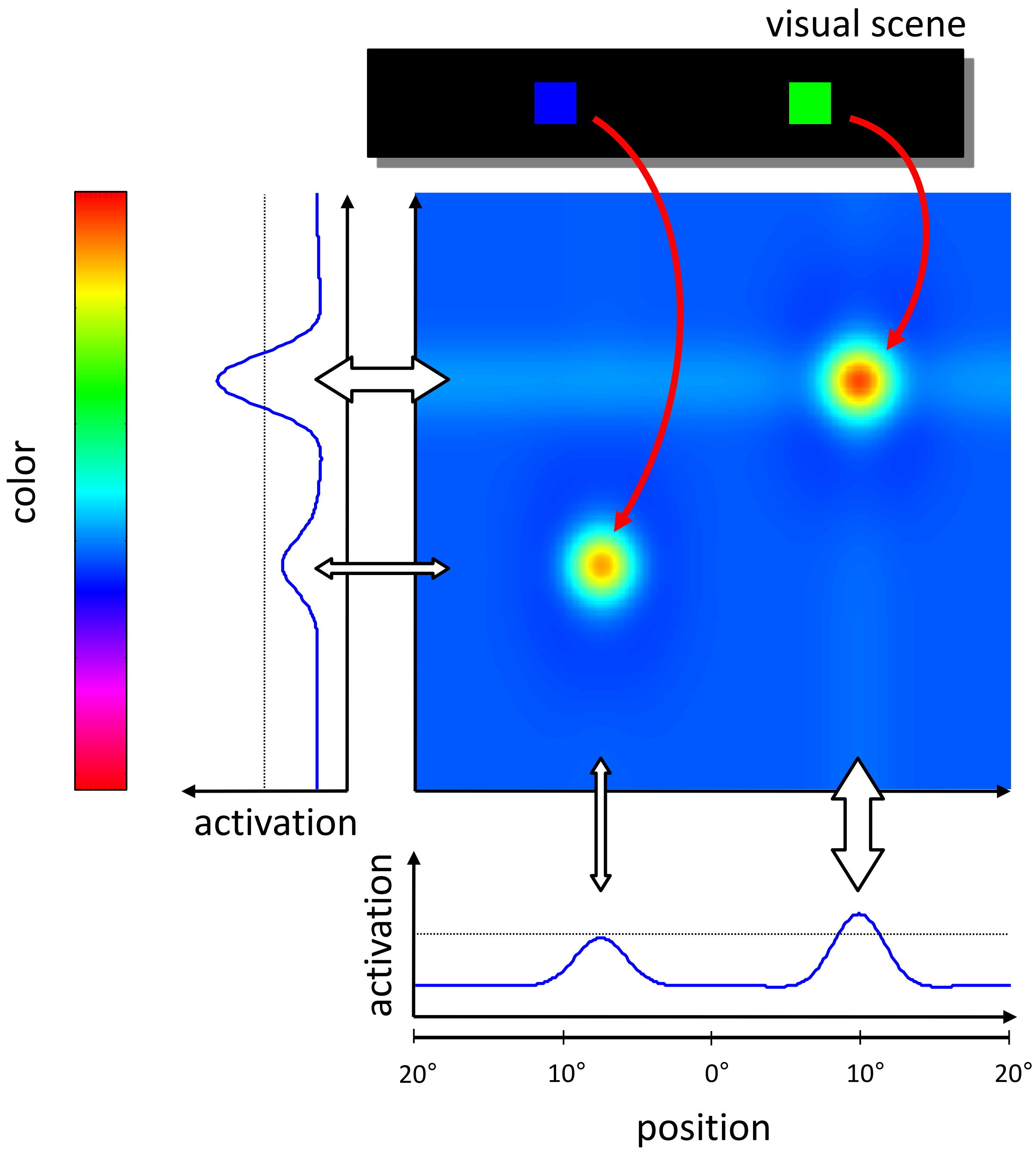

- if multiple items match the specified feature value, all are amplified
- gradual strengthening for partial match
- further operations on output dependent on task: selection or representations of multiple results

# Joint selection with bidirectional projections




- bidirectional projections allow coupled selection in 1D fields
- can be biased by input to either 1D field

# Joint selection with bidirectional projections

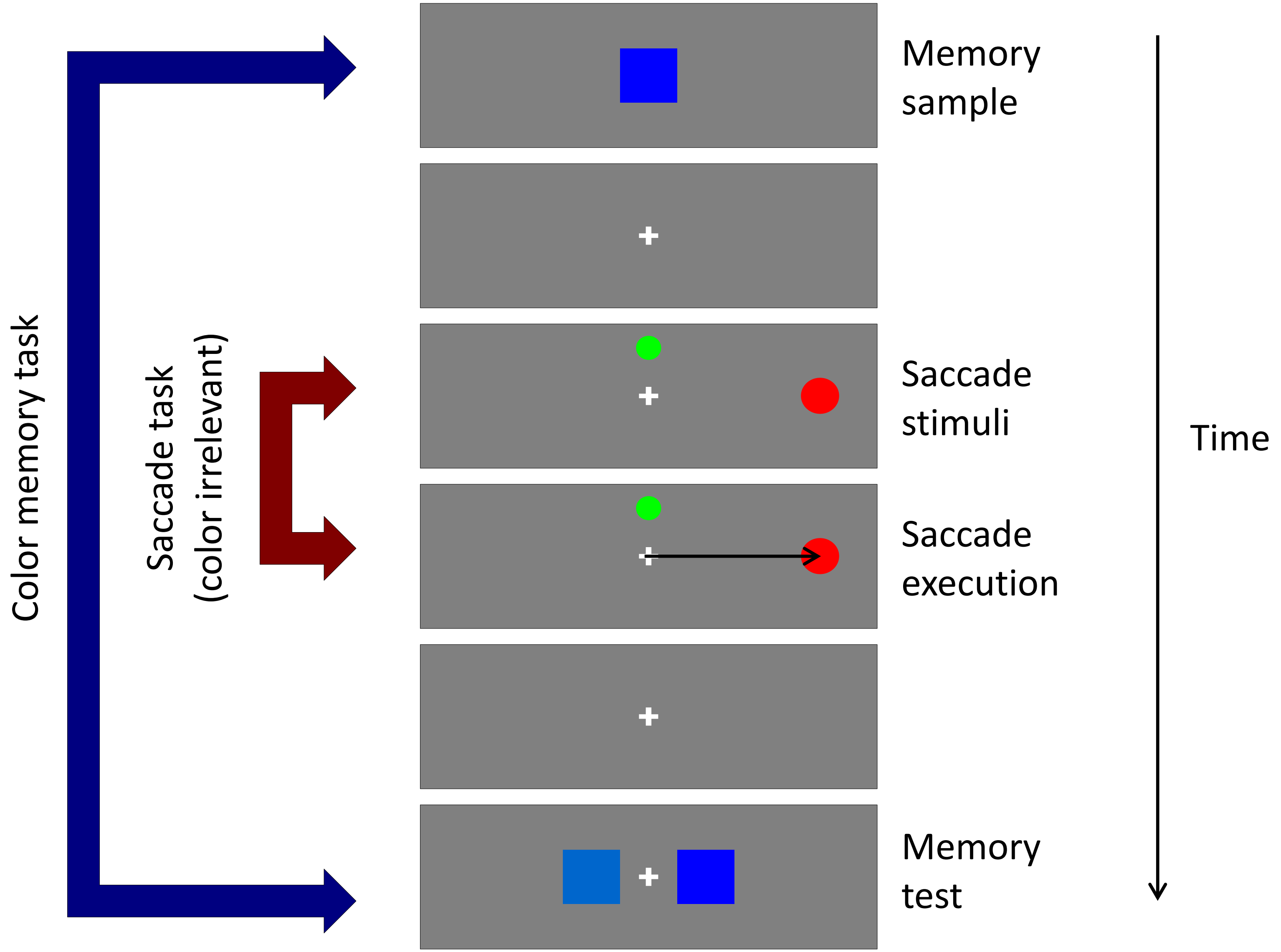



- once a single item is selected jointly in both 1D fields, ambiguity in feature conjunctions is resolved
- object features can then be processed in separate pathways
- sequential processing for multiple items

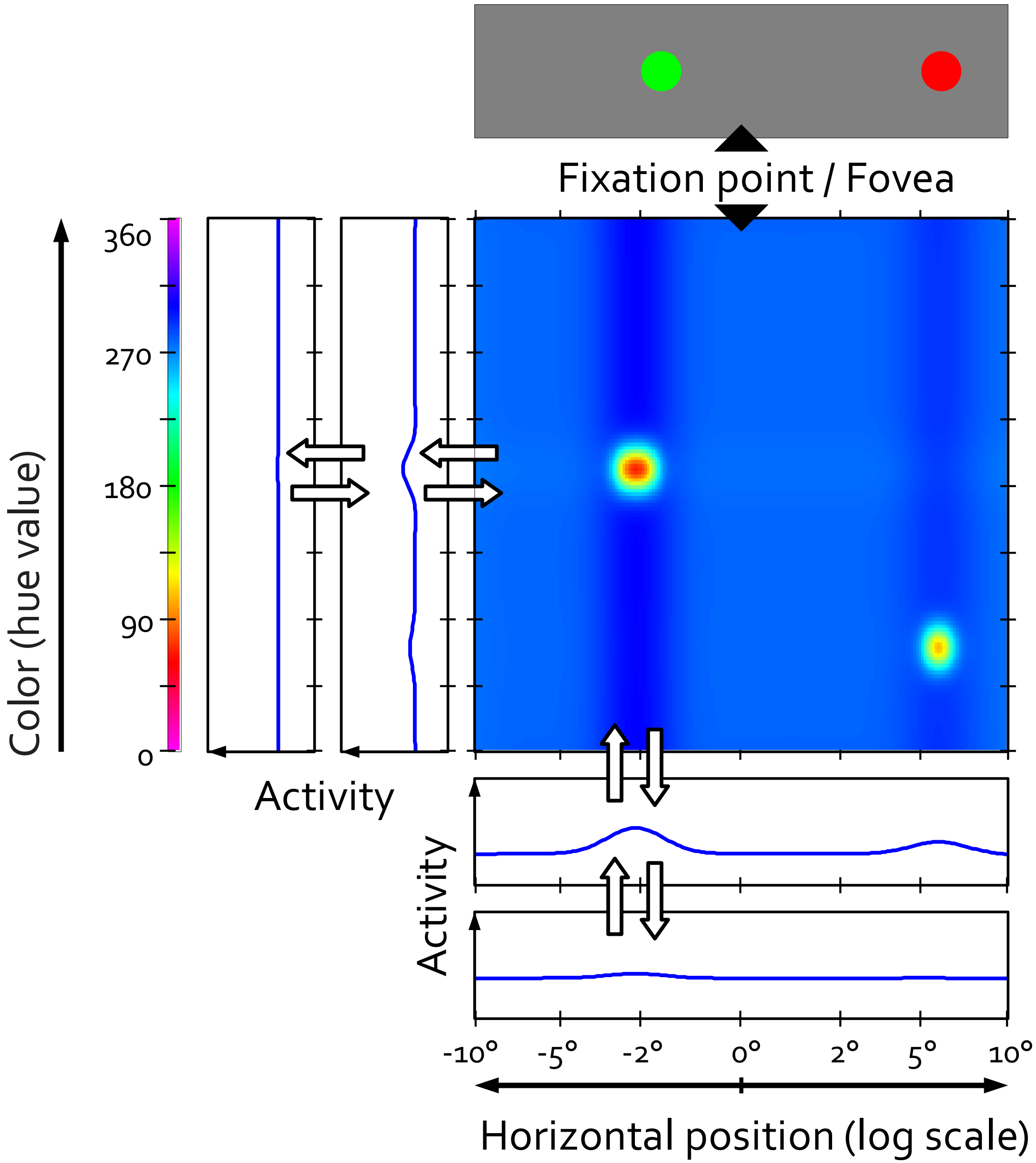
# Ridges and feature associations

Demonstration: ridge inputs, peaks from intersections, associations

# Case Study: VWM Biases Saccade Behavior



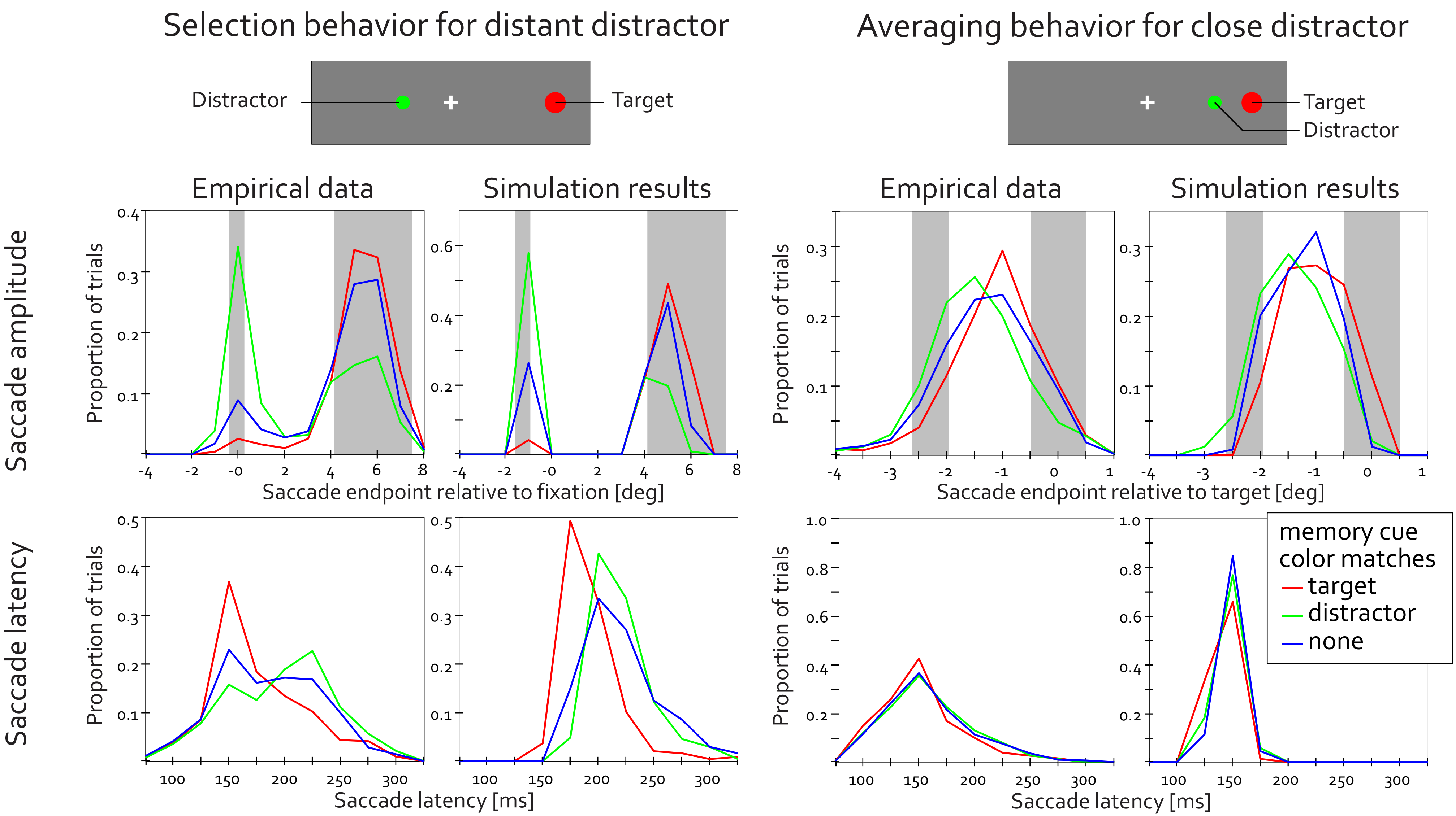
# Case Study: VWM Biases Saccade Behavior



# Case Study: VWM Biases Saccade Behavior

Video

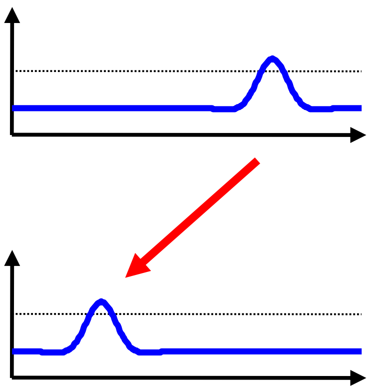
# Case Study: VWM Biases Saccade Behavior



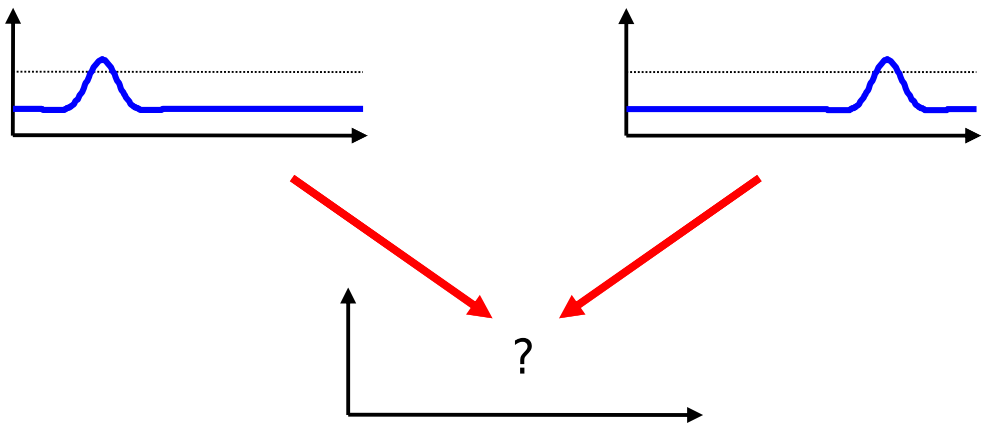

[Schneegans, Spencer, Schöner, Hwang, Hollingworth, in preparation]

# Operations in higher-dimensional fields

- projections between fields can implement simple mappings if they meet certain conditions (e.g. continuity)

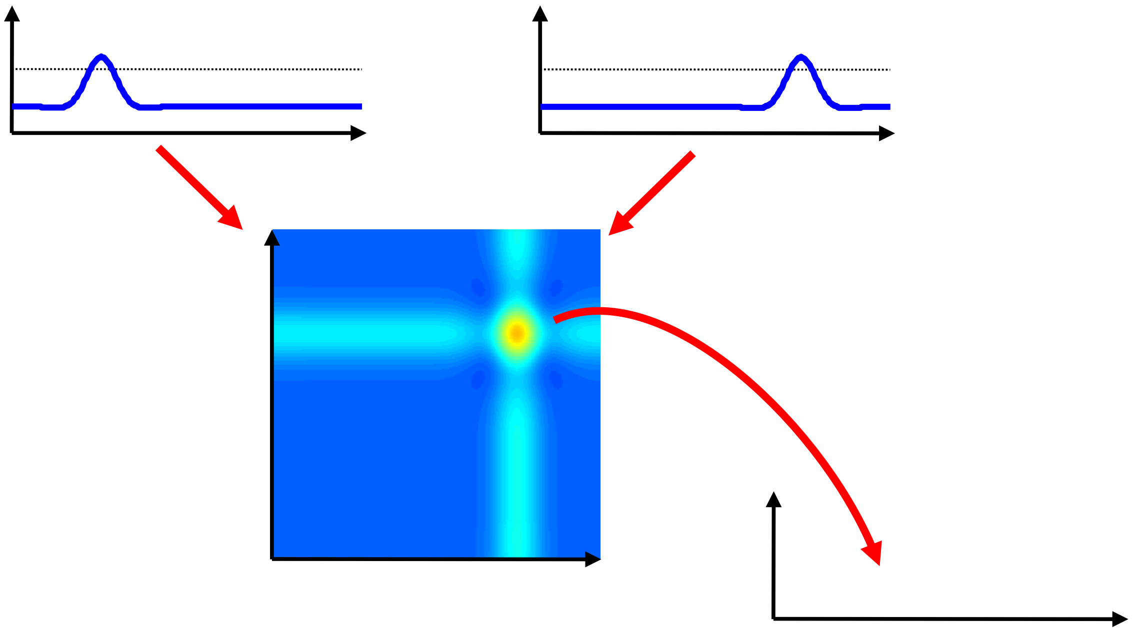


- what about operations that combine two different inputs? projections from two sources to a common field only allow additive combination of inputs

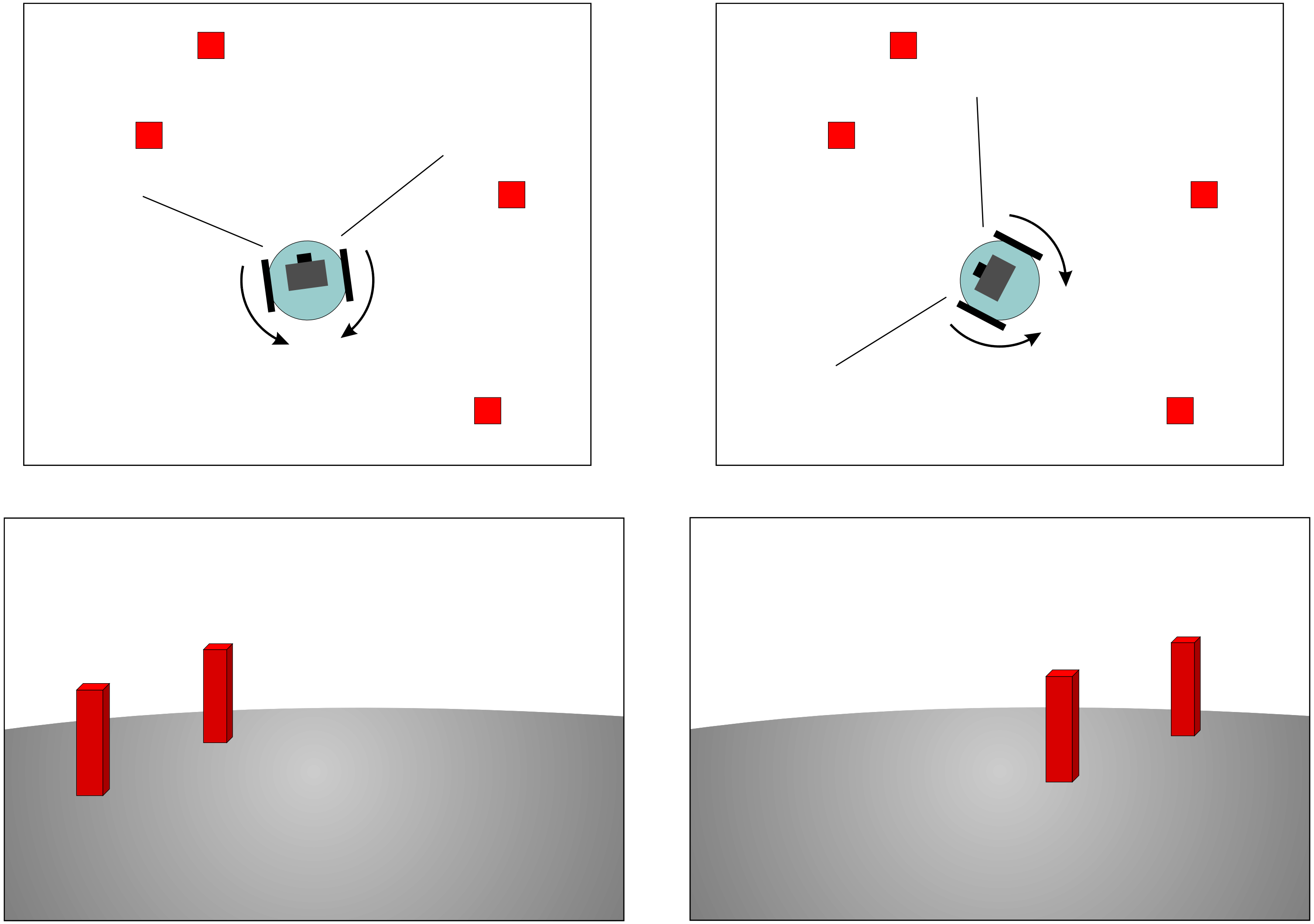


# Operations in higher-dimensional fields

- combining/expanding representations into a single high-dimensional field allows arbitrary mappings to an output field (as long as mapping is continuous)



# Example: Reference frame transformation



- eyes/camera only provide spatial information in retinal reference frame
- for orientation, gaze-invariant locations often more relevant

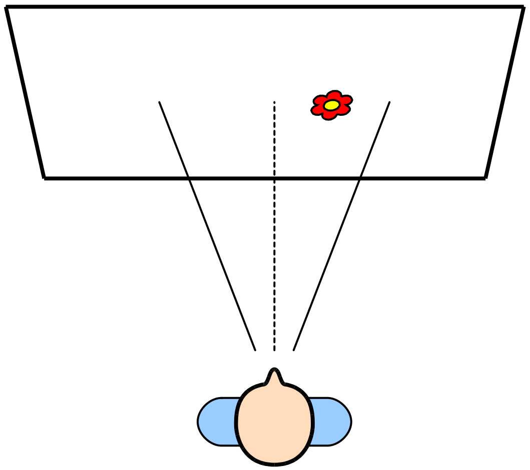
# Reference frame transformations

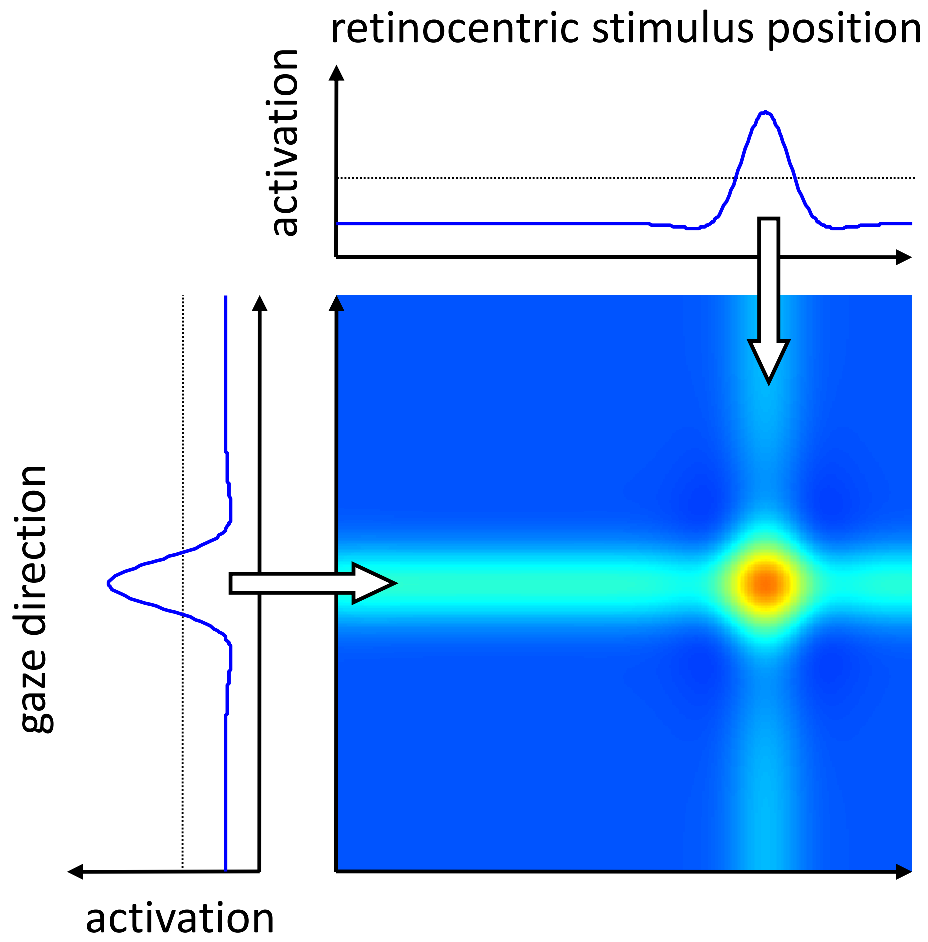
- gaze-invariant (e.g., body-centered) location information can be determined from combination of gaze direction and retinal locations
- for pure rotation and angular positions, reference frame transformation is simple shift
- can be expressed arithmetically as addition of vectors:

$$p_{inv} = p_{cam} + p_{gaze}$$

- but DNF representations based on population codes instead of numeric vectors, and fixed synaptic connections instead of arithmetic operations

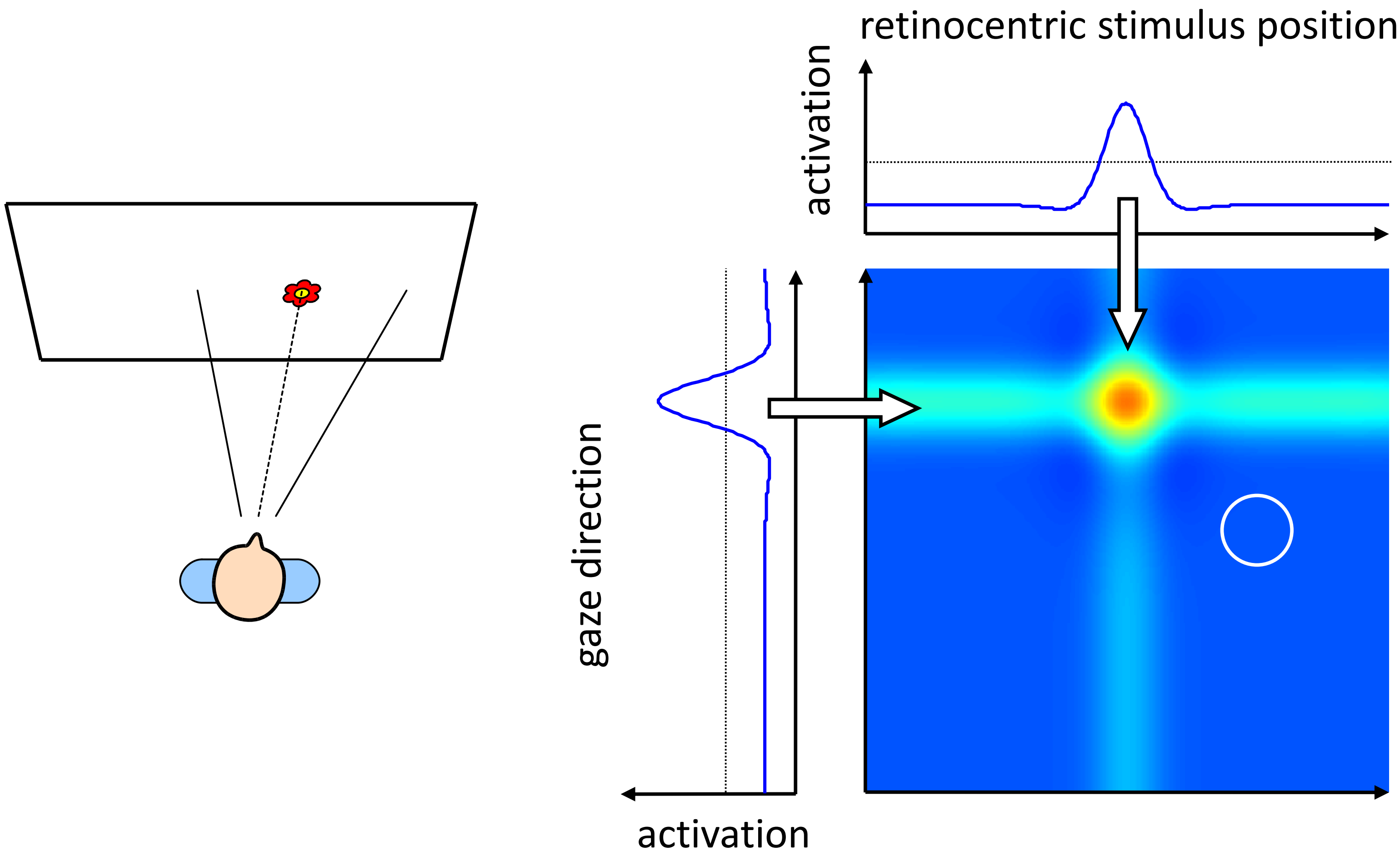
# Spatial transformations



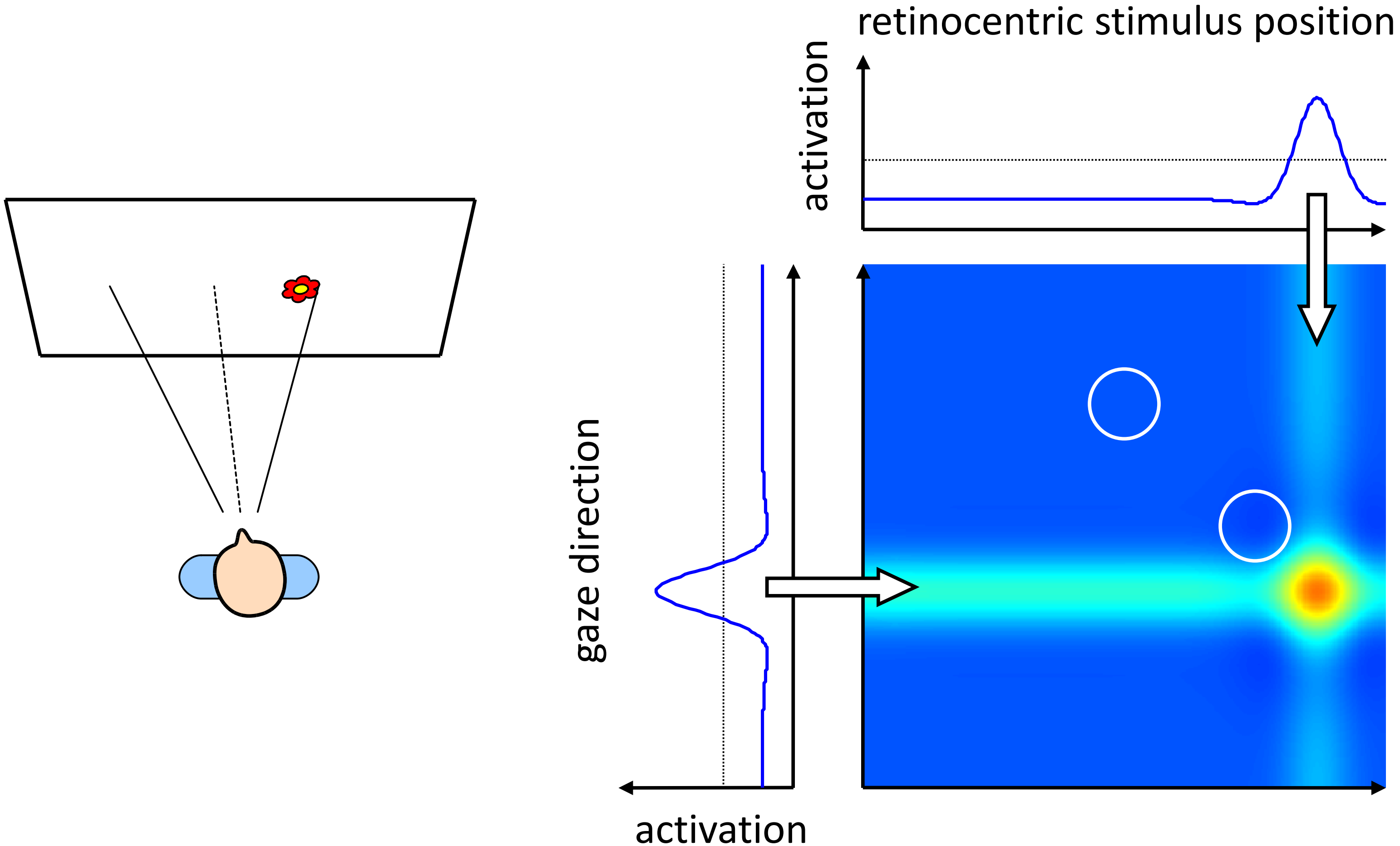



- for transformation of 1D location information: 2D field over retinal space and gaze direction

# Spatial transformations

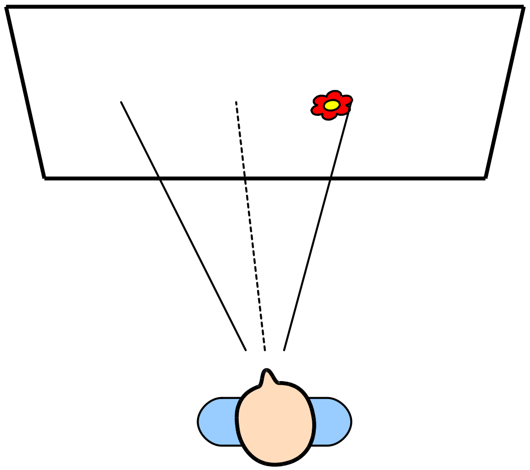


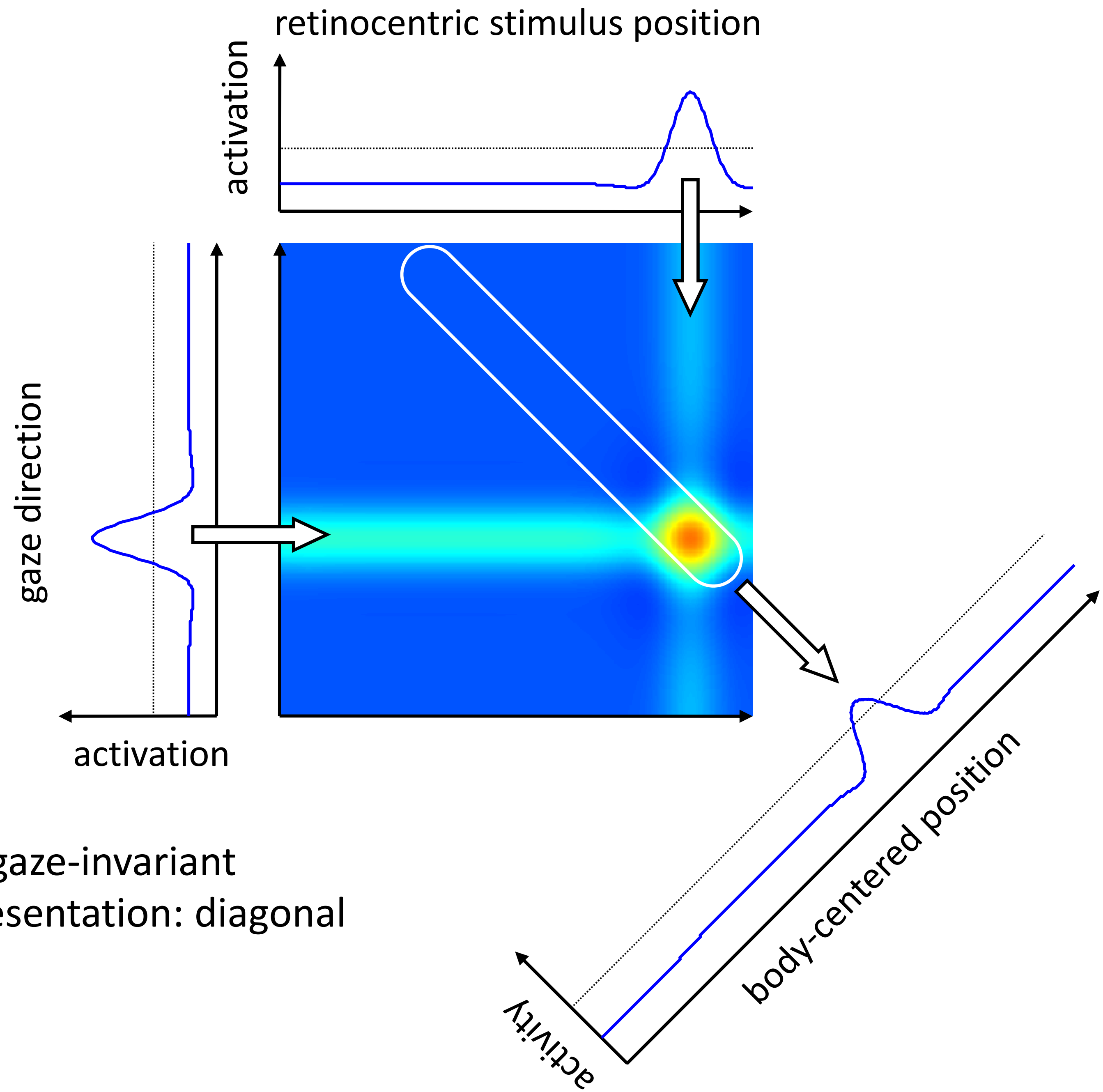
# Spatial transformations




- in angular coordinates for pure rotations: ego-centric stimulus position shifts by inverse of orientation change
- → points corresponding to the same location lie on a diagonal in the combined representation

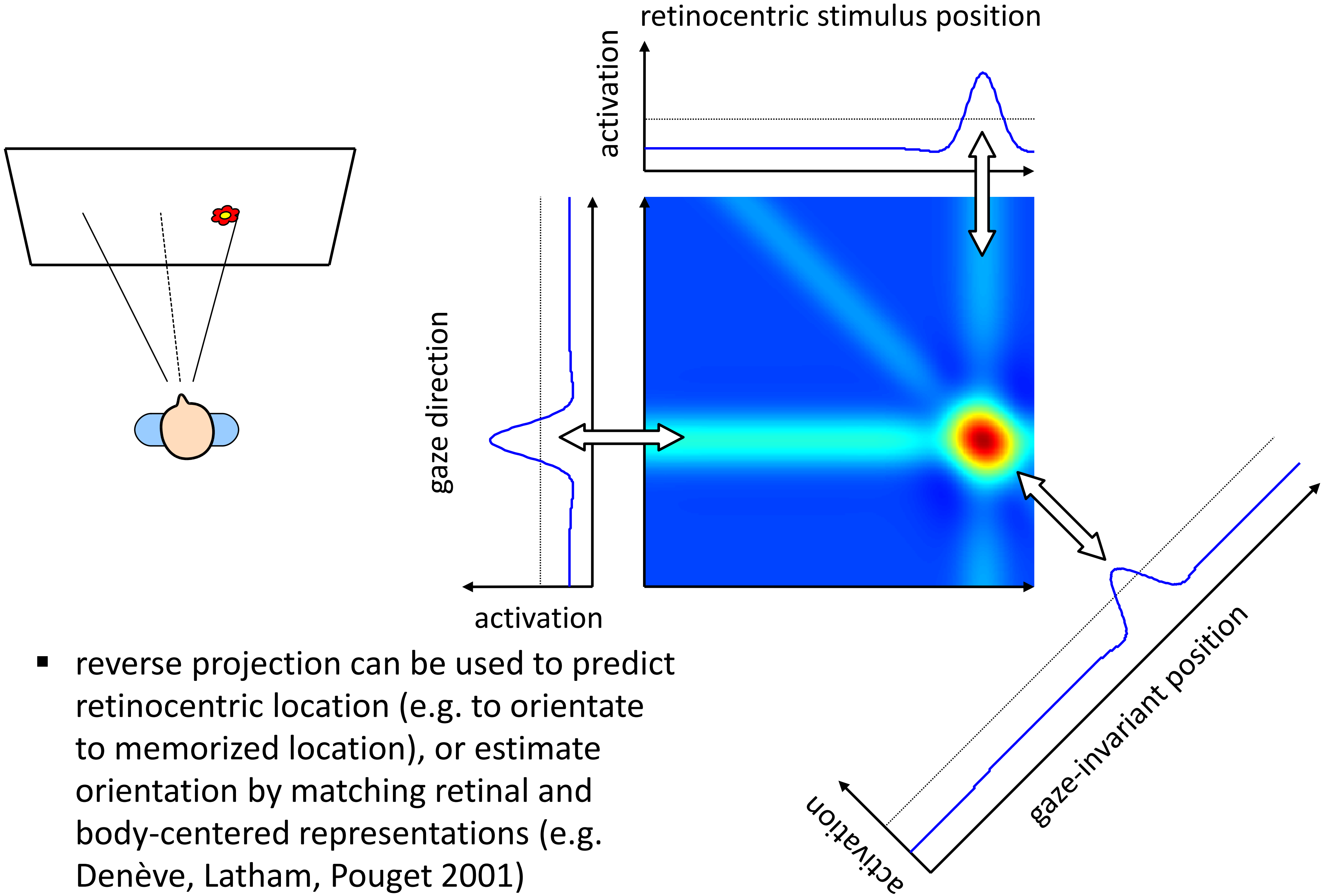
# Spatial transformations






- can be mapped onto gaze-invariant (body-centered) representation: diagonal read-out

# Spatial transformations

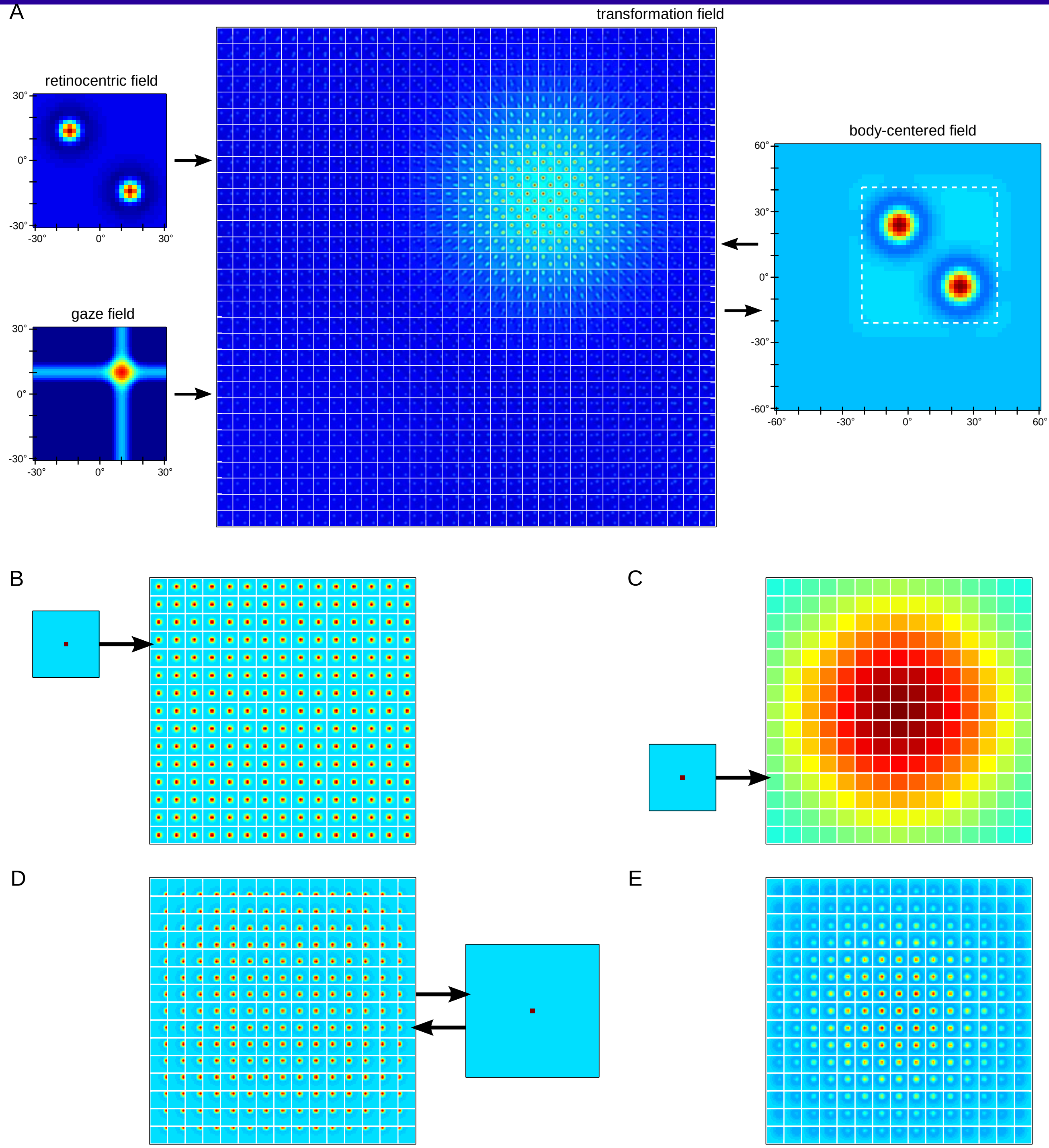



- reverse projection can be used to predict retinocentric location (e.g. to orientate to memorized location), or estimate orientation by matching retinal and body-centered representations (e.g. Denève, Latham, Pouget 2001)

# Case Study: Saccadic Remapping Model

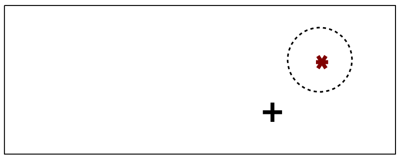
Video

# Case Study: Saccadic Remapping Model

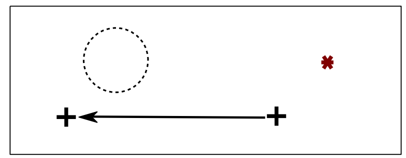


# Case Study: Saccadic Remapping Model

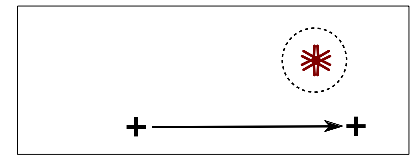
## Condition



Stimulus in RF turned on and off

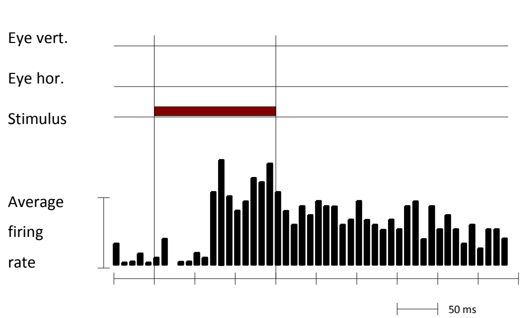


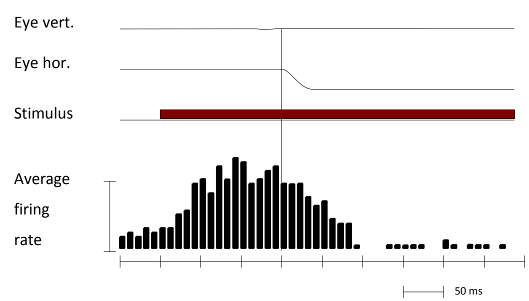
Saccade moves stimulus out of RF

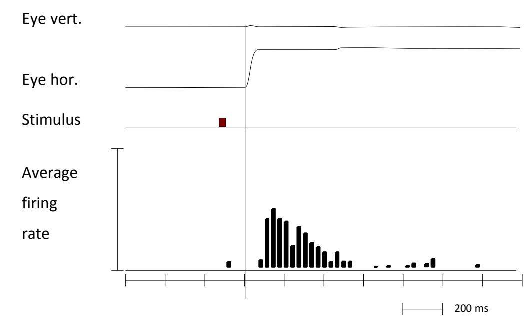


Saccade brings former stimulus position into RF

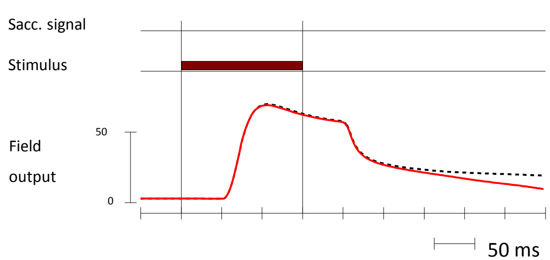
## Experimental results (average spike rate of single-cell recording in LIP)

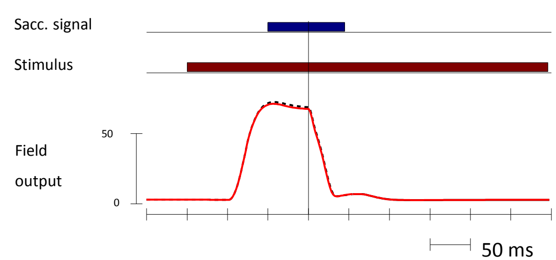


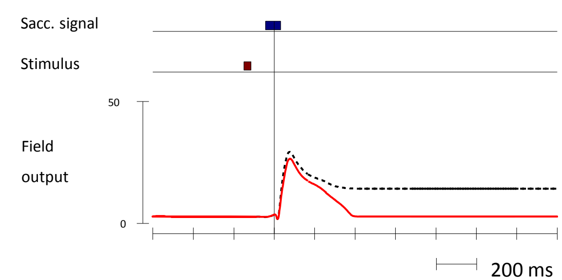




## Simulation results (field output at one retinocentric position)

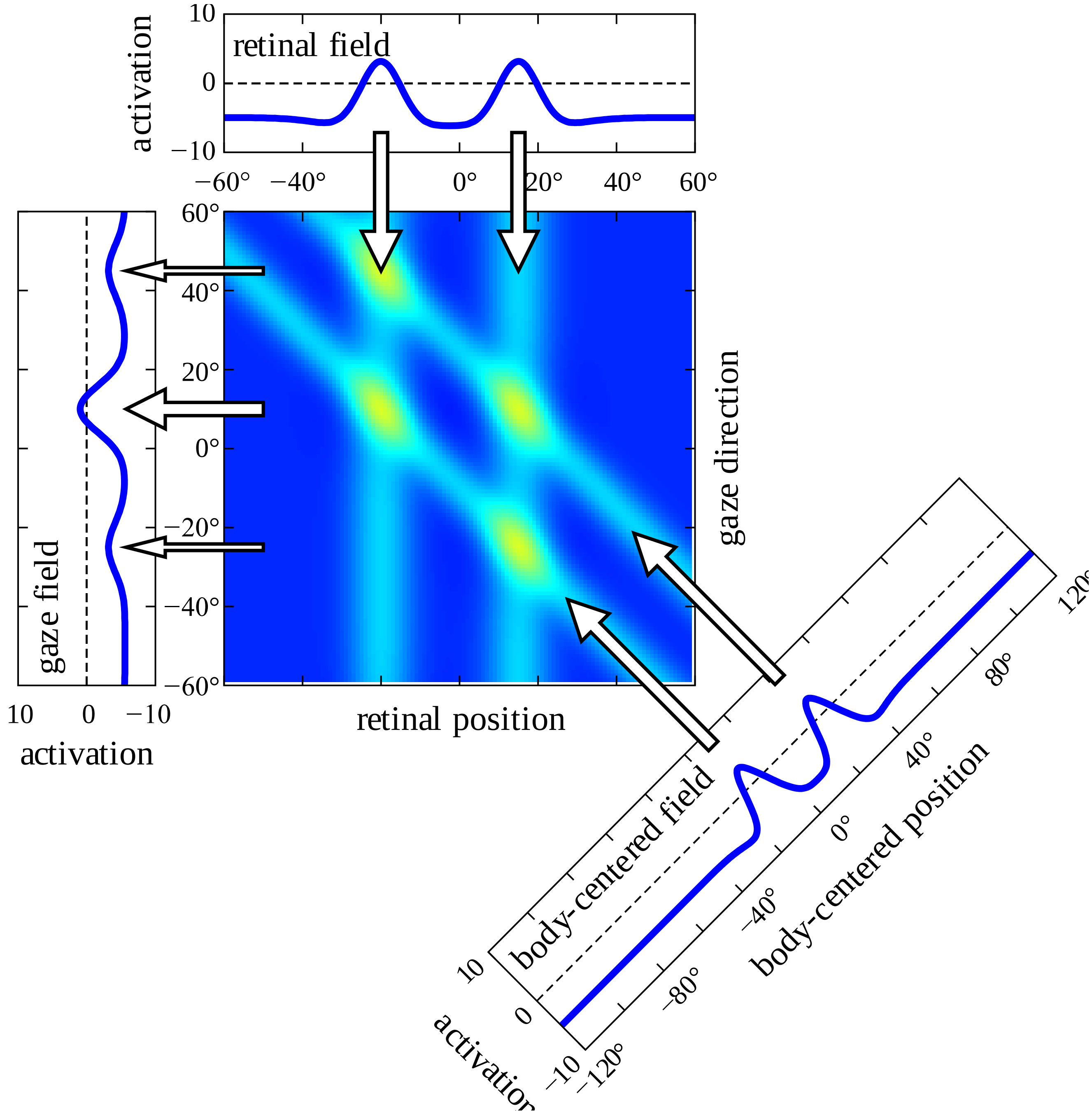



[Schneegans, Schöner 2012; experimental results by Duhamel et al. 1992]

# Spatial alignment for orientation estimation



# Conclusions

- higher-dimensional fields can represent multiple feature dimensions in a combined fashion
- more costly than low-dimensional fields, but needed to represent feature conjunctions rather than separate feature values
- associations between feature dimensions via higher-dimensional fields, e.g. for visual search
- higher-dimensional fields can implement complex mappings between feature dimensions, e.g. for spatial transformations