

A neural dynamic architecture for visual search

DFT Summer School 2021

Basic Information

The goal of this project is to implement a dynamic field architecture that is able to find and attend to objects with a given color in the visual input.

The architecture receives input from a camera or an image. It further receives a color as input, which is provided by activating a neural node that represents that color. The model then finds objects in the visual input with the activated color, ultimately generating peaks on their spatial locations in a *spatial attention field*.

Figure 1a depicts an example image. Figure 1b shows peaks on all blue objects in the *spatial attention field*.

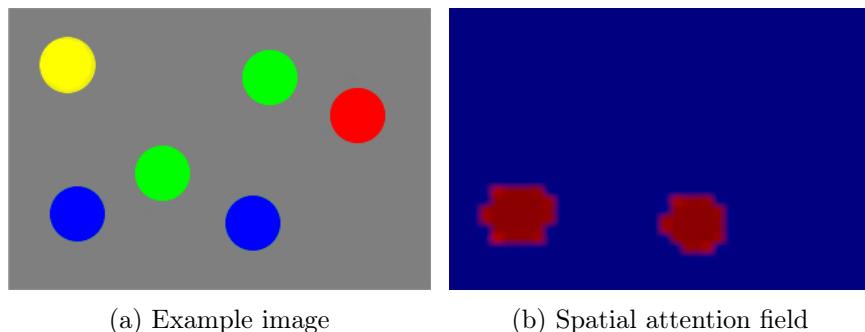


Figure 1: Finding all blue objects in an example image

Figure 2 shows a schematic depiction of the architecture. In the following, we will go through each of the components in detail, and you will be asked to implement them in cedar.

Visual preprocessing

The perceptual system is comprised of fields that represent objects in terms of their color. Please open the provided *template.json* in cedar. This file already contains a visual preprocessing pipeline. At the very left of that pipeline, you find a PICTURE component that outputs a static picture file. Please right-click on the PICTURE component and click on “plot all” to view the currently loaded picture. You can change the picture anytime by selecting a different filename in the settings inspector.

Next, please right-click on the “Image source” group and click on “plot all” to view the output of the visual preprocessing pipeline. That output is an activation distribution over a three-dimensional space, comprised of the two spatial dimensions of the visual input and an additional hue dimension. The activation scales with the saturation of the image at that position. In the window that you just opened, different hue values are depicted as different images, where each image depicts the saturation at that hue value for different spatial locations. For example, the image 24 (lower right; start counting in the upper left at 0) depicts the saturation in the red region

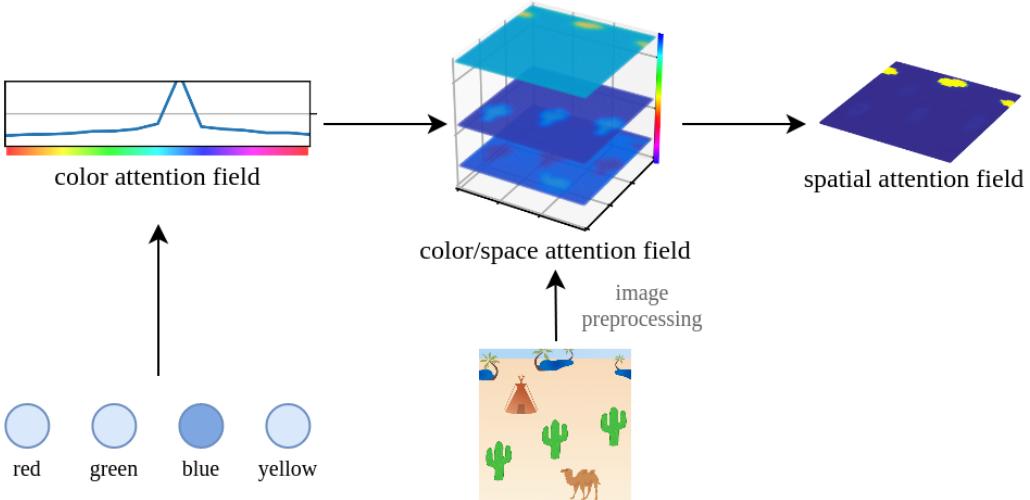


Figure 2: Schematic depiction of the model architecture.

of the color space, which has high saturation at the position of the red object. Image 7 depicts the saturation in the green region of the color space, which has high saturation at the positions of the two green objects.

Visual search

Your first task is to implement an attentional guidance mechanism that is able to select objects with a given color. The first part of this mechanism is a *color/space attention field*, which is a field that is defined over the two spatial dimensions of the visual input and the hue dimension, in the same way as the output of the visual preprocessing. Please create such a field by dragging and dropping the NEURALFIELD component, and connect the output of the visual preprocessing to that field. Note that you need to choose the sizes of the 3 dimensions in accordance with the sizes of the visual preprocessing output, which you can find out by moving your mouse over the output dot to the right of the visual preprocessing group. Please choose the field parameters in such a way that only subthreshold bumps of activation are created.

The second part are color nodes, which are neural nodes that signify that attention should be guided to objects of a certain color. Create neural nodes for the colors RED, GREEN, BLUE, and YELLOW by dragging and dropping the NODE component. Further, create and connect BOOST components that allow you to activate these nodes on demand.

The third part is a *color attention field*, which is a one-dimensional field defined over the hue dimension. Create such a field. Note that it has the same size as the hue dimension of the *color/space attention field*.

Upon activation of a color node, a peak should form in the corresponding region of the *color attention field*. For this purpose, the nodes have to project to the field with synaptic connection weights that correspond to a Gaussian centered on the location of the respective hue value. In cedar, this can be achieved using the HEBBIAN-CONNECTION component. Create such a component for each color, and connect the respective color node to its input “source node”. Set the target dimensionality and target size in accordance with the *color attention field*. Furthermore, select the option “manual weights”, which allows you to specify the synaptic connection weights as a parameterized Gaussian. Center that Gaussian on the respective hue value (a sigma value of 3 should work for all four colors). The amplitudes should slightly exceed the resting level of the *color attention field*. Finally,

connect the output of the component to the *color attention field*. Test the mechanism by activating different color nodes and making sure that peaks form in the appropriate regions of the *color attention field*.

When the *color attention field* forms a peak on a certain hue value, we want the *color/space attention field* to form peaks for all objects with that hue value. For this purpose, the *color attention field* has to provide slice input to the *color/space attention field* along the shared hue dimension. This is tantamount to increasing the dimensionality of the output of the *color attention field* to 3, and copying the value of c across the two added dimensions. For this purpose, create a PROJECTION component that receives input from the *color attention field*. Configure its output dimensionality and output dimension sizes in accordance with the *color/space attention field*. Select the index of the hue dimension for the dimension mapping property. After that, use a STATICGAIN to connect its output to the *color/space attention field*, and tune the gain factor so that the latter field forms peaks on objects with an activated color and no other objects. Test the mechanism by activating different color nodes and making sure that peaks form on the correct objects in the *color/space attention field*.

The last part is the creation of a *spatial attention field*. That field should be defined over the two spatial dimensions of the visual input, and should form peaks on the spatial locations of objects that receive attention in the *color/space attention field*. Try to figure out the steps to achieve this yourself. Note that you will need to implement a dimensionality reduction from 3 to 2 that sums the activation across the dropped dimension. This can be achieved with a PROJECTION component.

Test your architecture by trying to find and attend to objects of different colors with different input images. If you have a camera, you may want to replace the static picture by a CAMERA component, which allows you to perform visual search in any scene that you can point your camera to.

Congratulations! You have just implemented a neural dynamic architecture that comprises a simple model of visual search in the human brain. This architecture serves as a basis for many DFT architectures. An extension of this architecture that adds to this an interaction with scene memory, the attentional detection of visual transients, and the extraction of search cues, is described in [1]. Other extensions use this as a building block for grounding spatial language in perception ([2], [3], [4]). In the following, you can choose between one of three possible follow-up projects:

- A *neural dynamic architecture for goal-directed reaching*. In this project, you will build a model that enables a robot to reach for objects in its visual field.
- *Using serial order to learn and replay a sequence*. In this project, you will build a model for learning and reproducing sequences of colors.
- A *neural dynamic architecture for the perceptual grounding of simple spatial language*. In this project, you will build a model that can find and attend to objects matching a linguistic description like “the red object below the green object”.

References

- [1] Raul Grieben, Jan Tekülve, Stephan KU Zibner, Jonas Lins, Sebastian Schneegans, and Gregor Schöner. Scene memory and spatial inhibition in visual search. *Attention, Perception, & Psychophysics*, pages 1–24, 2020.
- [2] John Lipinski, Sebastian Schneegans, Yulia Sandamirskaya, John P Spencer, and Gregor Schöner. A neurobehavioral model of flexible spatial language behaviors. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 38(6):1490, 2012.

- [3] Mathis Richter, Jonas Lins, Sebastian Schneegans, Yulia Sandamirskaya, and Gregor Schöner. Autonomous neural dynamics to test hypotheses in a model of spatial language. In *Proceedings of the Annual Meeting of the Cognitive Science Society*, 36(36). Cognitive Science Society, 2014.
- [4] Daniel Sabinasz, Mathis Richter, Jonas Lins, and Gregor Schöner. Grounding spatial language in perception by combining concepts in a neural dynamic architecture. In *Proceedings of 42nd Annual Conference of the Cognitive Science Society*. Cognitive Science Society, 2020.