Class-Agnostic Robotic Gaze Control via Fast Normalized Cut

Gaze control for foreseeing robots

Abstract We present a class-agnostic approach to robotic gaze control using Fast Normalized Cut and DINOv3 visual features. The method partitions feature maps without explicitly constructing the pairwise similarity matrix defined by feature dot products, achieving linear scaling with the number of image patches per iteration. Recursive bipartitioning with a feature-similarity stopping criterion produces object masks without predefined object categories. These masks guide the robot's head movements to keep an object centered in its field of view. The system runs in real time on a gaming notebook with a GPU, demonstrating its applicability to interactive robotic perception. It also provides object descriptors that support distinguishing between observed objects. This way, the robot can see objects of arbitrary classes.

Fast Normalized Cut

Shi and Malik [1] designed a measure $NCut(A,B)$ for bipartitioning of pixels V into A and B ($A \cup B = V$, $A \cap B = \emptyset$) using a similarity measure $W_{p,q}$ among the pixels $p, q \in V$:

$$NCut(A,B) = \frac{sim(A,B)}{sim(A,V)} + \frac{sim(A,B)}{sim(B,V)} \quad \text{where} \quad sim(P,Q) = \sum_{p \in P, q \in Q} W_{p,q}$$

is a total similarity between the two sets of pixels P and Q. They proposed a spectral relaxation of minimizing $NCut(A,B)$, calculating the bipartition from the eigenvector corresponding to the second smallest eigenvalue of

$$D^{-\frac{1}{2}}(D-W)D^{-\frac{1}{2}}$$

where $D = W1^T$.

Wang [2] was the first to use Normalized Cut to bipartition the feature maps produced by deep learning models into object masks. He defined $W = T_\tau(FF^T)$ where F is the matrix of size $n \times m$, where n is the number of patches (regions on the image), and m is the number of features. T was a thresholding that kept the scalar products over a threshold $\tau > 0$.

Lucny [3] designed the algorithm that can calculate the bipartition y without materialization of W, defining $W = FF^T$, supposing such F that the corresponding R is well-defined and positive semidefinite. The method is based on finding the eigenvector corresponding to the largest eigenvalue of

$$U = D^{-\frac{1}{2}}WD^{-\frac{1}{2}} - \frac{z_0}{|z_0|}\left(\frac{z_0}{|z_0|}\right)^T \quad \text{where} \quad z_0 = D^{\frac{1}{2}}1_n$$

It is calculated by the power iteration using U for which

$$Uy = unproj(y, (d^{-\frac{1}{2}}1_m^T) \circ F) - unproj(y, \frac{z_0}{|z_0|})$$

where $unproj(u,v) := v(v^T u)$

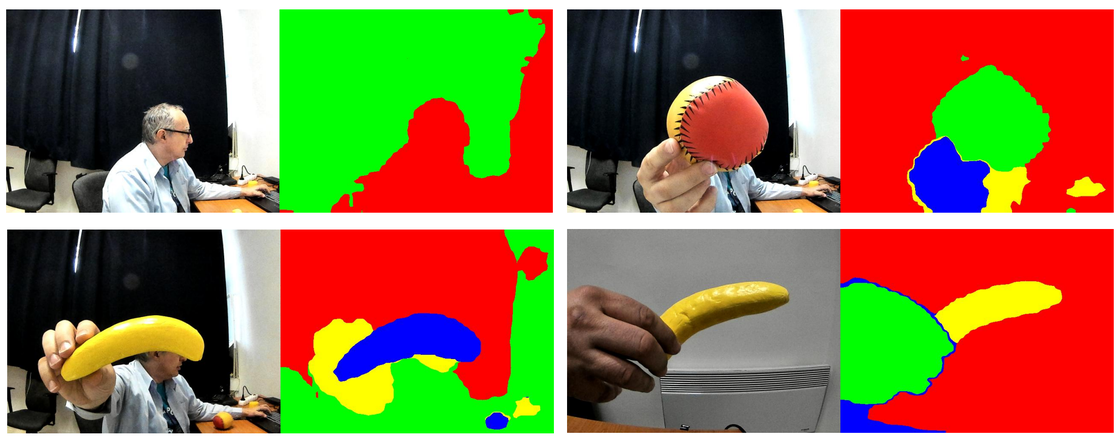
This approach scales linearly with the number of patches (or pixels). Moreover, the bipartition $y > 0$ converges much faster than the eigenvalue, so the number of necessary iterations is exceptionally low (1, 2, 4, ...).

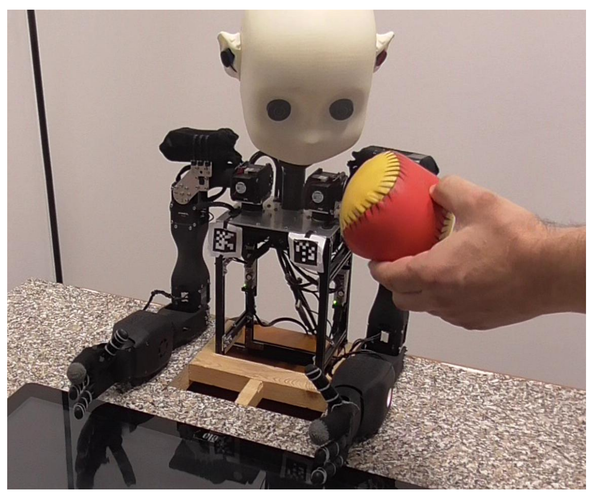
Robot's Gaze Control

When we run a foundation model on an image (we have used DINOv3 [4]), we get a feature map that describes the content of the image patches (384 features for 48×48 patches). Yet we can interpolate this feature map to a much finer resolution, e.g., 320×240. Then, since our bipartitioning algorithm is fast, we can run it recursively to split the image into a system of areas containing similar features. The problem is to determine whether we already have an object and avoid continuing its bipartitioning. The best results we have got with the rule that a component is split only if the cosine similarity between the average features of the two parts proposed by the bipartitioning algorithm meets:

$$\frac{\overline{F[A]} \cdot \overline{F[B]}}{|\overline{F[A]}||\overline{F[B]}|} < 1 - \tau d$$

where τ is a threshold (we used 0.05) and d is depth of the bipartition $(0, 1, 2, \dots)$





Achieving 6 fps on a gaming notebook, the control of a robot's head following objects we present is straightforward. The robot reacts to the presence of a complete object at the center of its field of view. We found that the algorithm is even too stable, and it would be convenient to add occasional random head movements to make it more human-like when no objects are present, or they are present but fixed. The contribution of this approach is that we can use really any object. Moreover, we obtain an object mask and a descriptor that enables us to distinguish among the objects.

[1] Shi, J., & Malik, J. (2000). Normalized cuts and image segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence, 22(8), 888–905.
[2] Wang, X., Girdhar, R., Yu, S. X., & Misra, I. (2023). Cut and learn for unsupervised object detection and instance segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 3124–3134).
[3] Lúčny, A. (2026). A fast algorithm for normalized cut with applications on bipartitioning feature maps in deep learning. SSRN
[4] Siméoni, O., et al. (2025). DINOv3: Self-supervised learning for vision at unprecedented scale. arXiv. https://doi.org/10.48550/arXiv.2508.10104

FAKULTA MATEMATIKY, FYZIKY A INFORMATIKY
Univerzita Komenského v Bratislave

MATFYZ CONNECTIONS

Andrej Lúčny
Department of Applied Informatics
lucny@fmph.uniba.sk
ORCID 0000-0001-6042-7434

Branislav Zigo
Department of Applied Informatics
Branislav.zigo@fmph.uniba.sk
ORCID 0009-0001-8216-2868

Igor Farkaš
Department of Applied Informatics
Igor.farkas@fmph.uniba.sk
ORCID 0000-0003-3503-2080

Funded by the EU NextGenerationEU through the Recovery and Resilience Plan for Slovakia under the project No. 09I01-03-V04-00048 (HUROSI).
Presented at ICRA 2026 in the poster section of the Late Breaking Results