

## Image Segmentation in Twenty Questions

Christian Rupprecht<sup>1,2</sup>, Loïc Peter<sup>1</sup>, Nassir Navab<sup>1,2</sup>

<sup>1</sup>Technische Universität München, Munich, Germany <sup>2</sup>Johns Hopkins University, Baltimore MD, USA

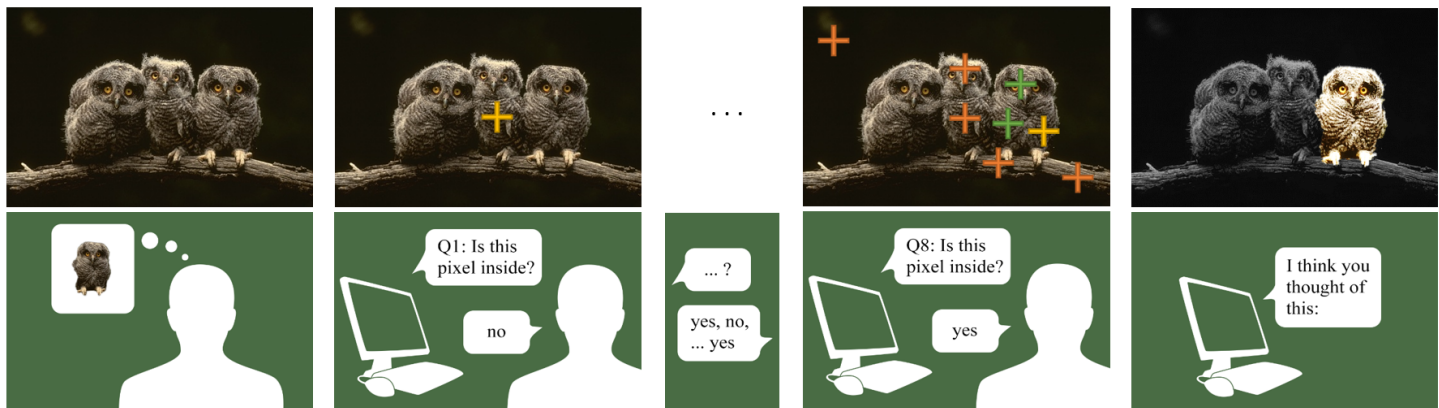


Figure 1: **Overview of the Twenty Questions segmentation scenario.** Given an image, the object to segment is secretly chosen by the human user. At every step, the computer asks whether a certain pixel is located inside the desired segmentation. After a predefined number of questions, it returns its guess about the answer.

Twenty Questions is a classical two-player game involving a questioner and an oracle. Before the game starts, the oracle thinks about something (e.g. an object, an animal or a character) which we will call the answer, and the questioner is allowed to ask a series of binary questions to guess what the answer is. While the game originally involves a human in the role of the oracle and another human as player, the development of artificial intelligence techniques led to softwares where the computer tries to guess what the human user has in mind.

Going in the same direction, we consider in this work the case of a computer playing the Twenty Questions game with a human user in the role of the oracle, where the expected answer is a *binary segmentation* of a given image. We introduce a strategy for the computer to provide a guess as accurate as possible in a limited number of questions. Alternatively, the proposed approach can also be seen as an interactive segmentation task. The crucial characteristic of our scenario is that the user interaction with the machine is restricted to binary inputs (*yes/no*) instead of the usual scribbles or bounding boxes [1]. This setting is common to provide a relevance feedback for interactive image retrieval [3, 5, 6] but was never considered in the context of segmentation. Moreover, if combined with e.g. a voice recognition system, our method can eventually provide a hands-free segmentation technique, for which an alternative solution was proposed by Sadeghi et. al. [4] who used an eye tracker for seed placement. This has potential applications like segmentation of medical data in sterilized operating room environments, where physical interactions with objects have to be avoided.

Our method can be summarized as follows. At each turn, the question to be posed consists in asking whether a well-chosen pixel is inside the hidden segmentation or not. The choice of the location to ask about is made as to maximize the information brought by the answer. More precisely, we define for any possible segmentation a probability to have been picked by the oracle. Each segmentation score is based on two aspects: its intrinsic quality in terms of image partitioning (i.e. is the segmented object rather homogeneous), and its compatibility with the answers already collected. Hence, this defines a probability distribution over the set of possible segmentations  $\mathcal{S}$ . In theory, the ideal question choice would follow a divide and conquer approach and halve  $\mathcal{S}$  in two parts of approximately equal probabilities. However, the huge size of  $\mathcal{S}$  excludes any exhaustive computation of the aforementioned distribution, which we approximate instead by sampling a series of segmentations proportionally to their probability of having been chosen by the oracle. This sampling is performed via a Markov Chain Monte Carlo (MCMC) framework, which builds on the Metropolis-Hastings

algorithm [2] and consists in exploring  $\mathcal{S}$  while spending more time in areas of high probability. The most informative question can be computed from the drawn samples in a tractable way, and the knowledge brought by the answer is included in the probabilistic model for the future rounds. We evaluate our method on three publicly available segmentation datasets with various visual properties and compare it with several baselines. The experiments demonstrate that our approach is a promising solution to the problem of interactive object segmentation with binary inputs only. The question selection strategy is fast enough (around 1 second) to be incorporated in a software and does not rely on any offline training step, which also makes the framework overall very adaptive to different types of images.

We introduce a method able to guess a segmentation in an arbitrary image by asking simple binary questions to the user. The questions are computed by approximating a posterior probability distribution over the set of segmentations with a MCMC sampling algorithm building on geodesic distance transform segmentation. Our method shows to be tractable for practical use with a waiting time between two questions of less than one second. No assumption about the type of the given image is necessary as the framework does not rely on any offline training step. The experimental evaluation performed on three very different datasets demonstrates that the approach provides an overall efficient solution to this problem.

- [1] L. Grady. Random walks for image segmentation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 28(11):1768–1783, Nov 2006.
- [2] Nicholas Metropolis, Arianna W. Rosenbluth, Marshall N. Rosenbluth, Augusta H. Teller, and Edward Teller. Equation of state calculations by fast computing machines. *The Journal of Chemical Physics*, 21(6): 1087–1092, 1953.
- [3] C. Nastar, M. Mitschke, and C. Meilhac. Efficient query refinement for image retrieval. In *CVPR 1998*, pages 547–552, Jun 1998.
- [4] Maryam Sadeghi, Geoffrey Tien, Ghassan Hamarneh, and M Stella Atkins. Hands-free interactive image segmentation using eyegaze. In *SPIE Medical Imaging*, pages 72601H–72601H. International Society for Optics and Photonics, 2009.
- [5] Kinh Tieu and P. Viola. Boosting image retrieval. In *CVPR, 2000. Proceedings.*, volume 1, pages 228–235 vol.1, 2000.
- [6] Simon Tong and Edward Chang. Support vector machine active learning for image retrieval. In *ACM International Conference on Multimedia*, pages 107–118, New York, NY, USA, 2001. ACM.