Solving Multiple Square Jigsaw Puzzles with Missing Piece

Genady Paikin[1], Ayellet Tal[1]
[1]Department of Electrical Engineering, Technion—Israel Institute of Technology

Puzzle solving is important in many applications, such as image editing [1], biology [5] and archaeology, to name a few. This work focuses on puzzles with square pieces. The problem is introduced by [2], where a greedy algorithm, as well as a benchmark, are proposed. The algorithm discussed in [8] improves the results by using a particle filter. Pomeranz et al. [6] introduce the first fully-automatic square jigsaw puzzle solver that is based on a greedy placer and on a novel prediction-based dissimilarity. Gallagher [3] generalizes the method to handle parts of unknown orientation. Son et al. [4] demonstrate a considerable improvement for the case of unknown orientation, by adding "loop constraints" to [3]. Rather than pursuing a greedy solver, Sholomon et al. [7] present a genetic algorithm that is able to solve large puzzles.

Our work is inspired by [6]. However, it takes the next step and not only improves upon the state-of-the-art results, but also solves puzzles with additional challenges. In particular, it handles puzzles with missing pieces, unknown size, and unknown orientation of the parts. Moreover, it concurrently solves multiple jigsaw puzzles whose pieces are mixed together, where neither the size of the puzzles nor a priori knowledge regarding possible missing pieces is given. This is illustrated in Figure 1, where more than 5000 pieces that belong to five different puzzles, some with many missing pieces, are given, and our algorithm reconstructs them faultlessly.

Like [3, 6], our algorithm is greedy. However, it incorporates three key ideas, which not only prove beneficial for solving traditional puzzles, but also support the additional requirements mentioned above. First, similarly to previous methods, our placement is based on the compatibility between pieces. We propose a more accurate and faster compatibility function, which takes advantage both of the similarity between the pieces, and of the reliability of this similarity. Intuitively, the reliability will be low for smooth areas, which should not be handled at early stages. Second, since greedy solvers are extremely vulnerable to the initial placement, we take special care when choosing the first piece. We require it to have distinctive borders and to be located in a distinctive region. Third, rather than choosing the best piece for a specific location, we select the piece that minimizes the likelihood of erring, regardless of its location. The idea is that the absolute location of a piece should be determined only when all the pieces found their neighbors. This is very similar to the way people solve puzzles.

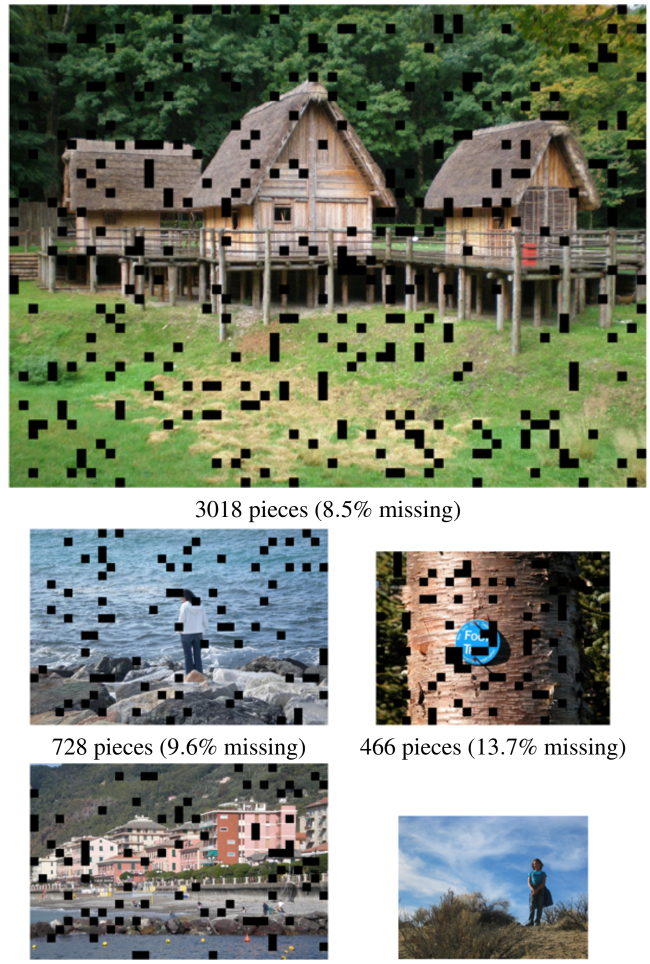
As a result, our algorithm is deterministic, conversely to previous approaches, which make random choices and are thus sensitive either to the selection of the first piece or to an initial random solution. This means that it suffices to run our algorithm only once.

To evaluate our algorithm, we apply our puzzle solver to the datasets of [2, 6, 7]. We first quantitatively compare our results to those of previous works [4, 6, 7] when applied to the traditional jigsaw puzzle problem, where the size of puzzle is known and there are no missing pieces. Our method outperforms the state-of-the-art methods, both in accuracy and in efficiency. For instance, it achieves accuracy of 97.7% on the dataset of [7], which consists of very large jigsaw puzzles, whereas [7] achieves 96.41%. This is done while accelerating the running times by a factor of 60-120.

Then we test our algorithm qualitatively on additional cases, including puzzles of unknown size, puzzles with missing pieces, and mixed puzzles. We show that our algorithm reconstructs the images well in various cases, including challenging mixed pieces from similar images.

Our contribution is hence a novel and efficient puzzle solver, which produces good results even when the input consists of pieces from multiple puzzles, each has missing pieces and unknown size. This may be the case not only in everyday situations, but also in archaeology, where torn documents or broken artifacts are found mixed and lack parts.

This is an extended abstract. The full paper is available at the Computer Vision Foundation webpage.



3018 pieces (8.5% missing)

728 pieces (9.6% missing) 466 pieces (13.7% missing)

723 pieces (10.2% missing) 432 pieces (0% missing)

Figure 1: **Solving multiple puzzles with mixed and missing pieces.** The input contains 5367 parts from five puzzles. Our algorithm has no knowledge regarding the size of the puzzles, the number of pieces per puzzle, or the number of missing pieces. Nonetheless, it reconstructs the puzzles accurately in 30 seconds.

[1] T.S. Cho, S. Avidan, and W.T. Freeman. The patch transform. *PAMI*, 32(8):1489–1501, 2010.

[2] T.S. Cho, S. Avidan, and W.T. Freeman. A probabilistic image jigsaw puzzle solver. In *CVPR*, 2010.

[3] A.C. Gallagher. Jigsaw puzzles with pieces of unknown orientation. In *CVPR*, 2012.

[4] K.Son, J. Hays, and D.B. Cooper. Solving square jigsaw puzzles with loop constraints. In *ECCV*, 2014.

[5] W. Marande and G. Burger. Mitochondrial dna as a genomic jigsaw puzzle. *Science*, pages 318–415, 2007.

[6] D. Pomeranz, M. Shemesh, and O. Ben-Shahar. A fully automated greedy square jigsaw puzzle solver. In *CVPR*, 2011.

[7] D. Sholomon, O. David, and N.S. Netanyahu. A genetic algorithm-based solver for very large jigsaw puzzles. In *CVPR*, 2013.

[8] X. Yang, N. Adluru, and L. Latecki. Particle filter with state permutations for solving image jigsaw puzzles. In *CVPR*, 2011.