

Benchmarking Learning Algorithms for Dexterous Multi-Arm Insertion of Semi-Deformable Objects

Konstantinos Chatzilygeroudis, Kungpeng Yao,
Bernardo Fichera, Farshad Khadivar, Ilaria Lauzana, Anaïs Haget, and Aude Billard

[^]LASA Laboratory, Swiss Federal Institute of Technology in Lausanne - EPFL, Switzerland, {firstname.lastname}@epfl.ch

Motivation

Most of our everyday activities, such as the usage of tools to grasp small objects, require coordinated motion of the arms (e.g., reaching) or the hands (e.g., manipulation). Performing these tasks with one arm is often infeasible, mainly because the dexterity and flexibility required for such tasks is beyond a single arm's capabilities. Similarly, a single robot arm is simply incapable of meeting the requirements of such complex tasks. A dual or multi-arm robotic system, on the other hand, extends the flexibility and capabilities of a single robot arm. It allows, for example, highly complex manipulation of small and deformable objects that would otherwise be infeasible for single arm systems. Yet, we are far from having robotic systems that can robustly achieve such dexterous manipulation skills.

One, however, can envision a very broad range of applications in households and factories that can benefit from such strategies. Examples include, placing and closing lids, packaging boxes in pallets, inserting USB cable into sockets. Completing this type of tasks requires to localize both objects with respect to one another (lid on top of box, cable into tube) and to adapt forces and movements of the two arms in coordination. Insertion of semi-deformable materials is made particularly difficult as one cannot build an explicit model of the deformation and interaction forces. Machine learning provides a structured framework that can allow robots to learn difficult-to-model problems by using their previous experience, without explicit modeling of the task constraints and possibly taking advantage of noisy expert demonstrations.

This benchmark proposes an evaluation of machine learning algorithms on a difficult multi-arm insertion task that involves collaboration among the arms to manipulate a small and semi-deformable object.

Task Description

The benchmark task is learning how to complete part of the assembly of a hand-made Swiss watch (see Fig. 1) using a multi-arm robot system: inserting a small semi-flexible and irregularly shaped object into the correct groove. This task poses a few challenges to robotic systems, including (but not limited to):

- Placing an irregularly shaped object in the correct groove,
- Handling a deformable object,
- Multi-arm coordination:
 - One arm stabilizing the system, and the other(s) performing the manipulation,
 - Performing object manipulation that requires 2 tools.

This task is inspired by the Skill Acquisition in Humans and Robots (SAHR)¹ ERC project that is studying watchmaking through its training process as a means to understand how humans acquire dexterous skills and then transfer these skills to robotic systems. In more details, the simplified assembly of a

¹<https://sahr.epfl.ch/>

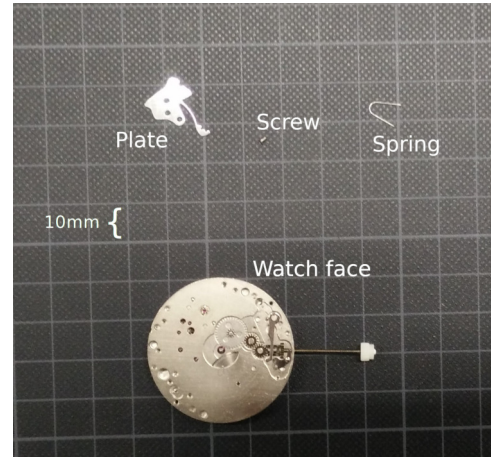


Fig. 1. The watch parts

Swiss watch consists of 3 main sub-tasks²: (1) Insert the spring in the correct location (Fig. 2), (2) insert the plate in the correct location (Fig. 3), and (3) mount the screw in order to fix the parts.

Both the spring and the plate are semi-deformable objects and it is required that you bend them in order to put them in place. Their small size additionally requires the usage of a second tool to keep the object stable. Fig. 4 shows the tools that are used in the actual setup: *the tweezers, a screw driver and a wooden tool*. In this benchmark, we focus on the plate insertion task. Thus, the tasks that need to be learned are:

1. Coordination in synchrony of the arms to orient plate for ease of control and successful insertion,
2. Change orientation and pressure in response to force feedback to insert the plate and its leg.

Task *One of the arms has the tweezers in its end-effector and the plate is already grasped (in a pre-defined location). Another arm has the second tool in its end-effector (either the wooden tool or a screw-driver). The task is to insert the plate in the correct location/groove on the watch face.*

Technical Details

Watch details The small size of the watch (e.g., the watch face diameter is $37mm$) creates additional challenges to an already difficult problem and makes it very hard to reproduce. For these reasons, we propose to use a *scaled-up version of the parts* and we have scaled the parts to around $3.5\times$ bigger than the actual one (see a prototype in Fig. 5).

Since the tweezers are required for our benchmark **we propose to add them to the YCB object database**. We believe that they will be useful for other researchers as they can be used in any fine-manipulation task. Additionally, the size of the scaled-up version *allows the usage of a screw driver instead of the wooden stick* and thus making the whole setup much easier to reproduce. Finally, we will make the screw driver compatible with the ones in the YCB object database or at least of canonical size.

²A video showing an expert performing the task can be found at: https://sahr.epfl.ch/wp-content/uploads/2018/10/Close-up-Video-for-Watchmaking.m4v?_=3

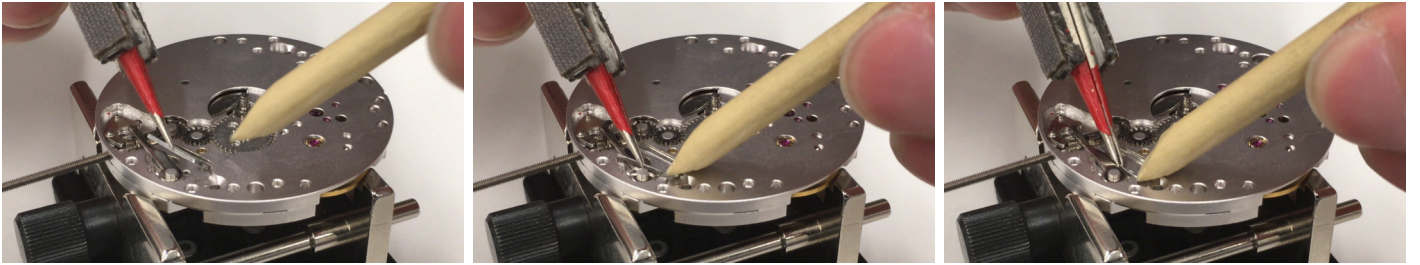


Fig. 2. Spring insertion

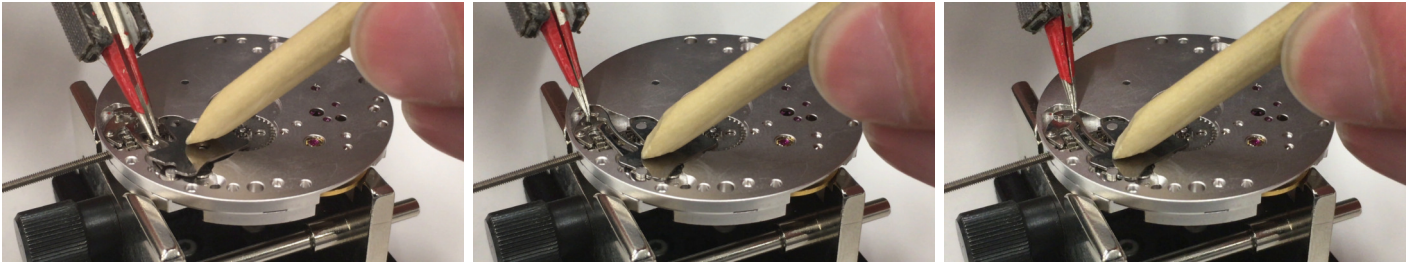


Fig. 3. Plate insertion



Fig. 4. The original tools

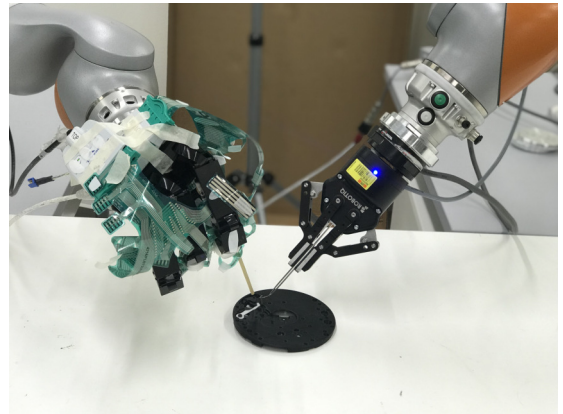


Fig. 6. An example setup



Fig. 5. A 3D printed prototype

Performance evaluation

This benchmark measures both the efficiency of the learning algorithms as well as the performance of the learned controller. The efficiency of an algorithm is measured in terms of interaction time; in other words, how much time it took to learn a successful controller. The performance of a controller is measured by evaluating how fast it can solve the task and its ability to handle multiple instances of the same problem (i.e., different initial locations of the watch face). The scoring is defined as follows:

$$sc = \frac{t_{max} - \frac{\sum_{i=0}^N t_i}{N}}{t_{train}}$$

where N (number of repetitions), t_{max} is the maximum time (in seconds) allowed to interact with the system for each repetition (controller execution), t_i is the interaction time (in seconds) needed to solve the task in i th repetition (if the task was not completed, this should be set to t_{max}), and t_{train} is the interaction time (in seconds) with the physical system needed for training. This scoring system promotes algorithms that have fast training times and that can robustly solve the same task with different initial watch face locations.

In essence, we let the learning algorithm run on the real system to find a controller. Once this controller is learned, we evaluate it by running it N times with different initial states each time.

Setup details The benchmark’s goal is to assess how fast and how robustly can learning algorithms learn to perform tasks that require multi-arm coordination and that are difficult to model (i.e., because of the semi-deformable objects). To this end, we disregard the computer vision challenges and we will assume that a computer vision algorithm based on colors or point-clouds and RGB-D sensors is available and can give the 6D poses of the objects. The detailed 3D models and point-clouds of the parts will be made available to facilitate this part. Nevertheless, because of the size of the parts, the tracking of the object is expected to have substantial uncertainty that the robotic system should be able to handle by incorporating haptic/force information from the interactions as well as the proprioceptive sensors of the robots. An example setup is shown in Fig. 6 where two KUKA IIWA robots are cooperating in order to perform a task similar to the one of the benchmark.