Proceedings of the ASME 2023
International Design Engineering Technical Conferences and
Computers and Information in Engineering Conference
IDETC/CIE2023
August 20-23, 2023, Boston, Massachusetts

DETC2023-XXXXXX

DEVELOPMENT OF A PILLOW PLACEMENT PROCESS FOR ROBOTIC BED-MAKING

Chi-Hong Cheung[1], Aaron Hao Tan[1], Andrew Goldenberg[1]

[1]University of Toronto, Toronto, ON, Canada

ABSTRACT

Bed-making is a common chore completed in various living environments to promote user comfort, hygiene, and well-being. Unfortunately, the physical and tedious nature of the act makes it challenging for segments of the elderly community to complete, and thus the opportunity arises to develop robots to automate the task. However, despite the opportunity's importance and positive impact, there is limited research on developing robotic bed-making systems. The aim of this research is to start addressing this gap by proposing methods to accomplish pillow placement, a major part of the bed-making task. This paper introduces a pillow placement process to be used by a static 6-DOF (degree of freedom) one-armed robotic manipulator with a 2-finger gripper. The process uses YOLOv4-tiny, image transformations, and principal component analysis (PCA) to infer pillow poses in a transformed RGB image, as well as a set of manipulator macro-actions to move pillows to their goal pose. We evaluated the proposed methodology in a real-world setting, where it enabled the robot to place pillows at desired poses on a miniature bed successfully in 89% of the experimental runs.

Keywords: Household robotics, applied machine learning, bed-making, pillow placement

1. INTRODUCTION

Chores are essential activities for creating a clean and organized living environment, which in turn promotes overall health and safety [1]. In long-term care facilities, caregivers aid the residents with tasks such as cooking [2], making phone calls [3], and bed-making [4]. However, the COVID-19 pandemic has made it challenging for caregivers to provide the required aid to this especially vulnerable population due to increased responsibilities and deteriorating mental health [5]. On the other hand, mobile manipulator robots have the advantage of not being susceptible to fatigue and diseases, characteristics that can be used to accomplish mundane tasks, surmount physical limitations, and alleviate the burden of chores [6, 7]. In fact, there are studies that suggest that older adults prefer having a robot assistant in cleaning,

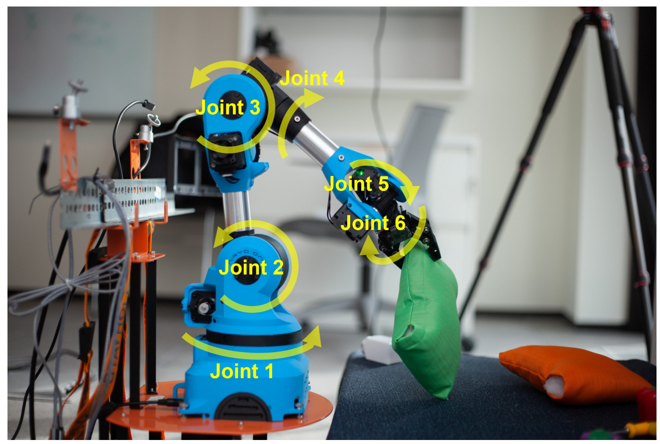

FIGURE 1: THE NIRYO ONE ROBOT WAS USED TO DEVELOP AND EVALUATE THE ROBOTIC PILLOW PLACEMENT PROCESS PROPOSED IN THIS PAPER. SUPERIMPOSED ON THE IMAGE ARE ARROWS AND LABELS THAT INDICATE EACH JOINT'S REVOLUTE MOTIONS.

fetching, and organizing tasks [7, 8]. One of the most physically demanding chores is bed-making, which involves preparing and arranging the bed for use. This task requires a certain degree of technical and practical skills [9, 10]. Specifically, the person must have the necessary flexibility, balance, coordination, and grip strength to perform the bending, reaching, and leaning motions of the task; these motions place physical loads on the body, increasing the risk of strain and injury [11]. Bed-making is especially challenging to segments of the elderly community who, due to their limited physical and cognitive ability, face barriers in accomplishing this tedious and strenuous task [12]. As the number of elderly who want to age at home increases [13], the development of solutions for aiding in chores such as bed-making is becoming increasingly urgent.

Bed-making is a complex multitask process [14] that includes rearranging the blanket, removing mattress wrinkles, and placing pillows. The list of tasks required to be fulfilled can further increase depending on user preferences (e.g. number of sheets), characteristics of the bed setup (e.g. bed placement in the room), and bed situation (e.g. needing to replace soiled linens). This research focuses on the pillow placement subtask, which involves placing pillows at their user-predefined desired pose on the bed, while addressing challenges in 1) pillow detection due to non-uniqueness and cluttered nature of an unmade bed, and 2) pillow manipulation due to the deformable nature of the pillows.

In this paper, we propose a novel process for a 6-DOF static one-armed robotic manipulator (Fig. 1) to address the challenges of pillow placement on a small-scale bed. Specifically, our main contributions are as follows: 1) we are the first to propose a unique process for robotic pillow placement that involves YOLOv4-tiny, image transformations, and PCA to estimate pillow position and orientation in varying bed scenarios, as well as an iterative sequence of manipulator pillow-placing macro-actions that are specially designed to deal with the deformable nature of pillows; 2) to address the challenge of the non-uniqueness and cluttered nature of bed-making scenarios, we constructed a novel RGB-D image dataset of diverse bed scenarios with which the YOLOv4-tiny model was retrained and orientation estimation method was developed; and 3) we performed real-world testing to evaluate the feasibility of the proposed approach in terms of percentage of pillows successfully placed.

2. RELATED WORK

Existing studies that inform the development of a pillow placement robot fall into 3 main categories: 1) Bed-Making Robotics [15–19], 2) Environment Understanding [20–25], and 3) Manipulation [26–30].

2.1 Bed-Making Robotics

The robotic bed-making task has been approached using both classical [15, 16] and learning-based methods [17–19]. Classical methods typically include using classical computer vision techniques (e.g. colour segmentation), coordinate transformations, and augmented grippers, and have been used to surmount specific challenges such as limited bed-making data availability. Namely, in [15], the authors proposed a method where the robot first chooses a grasping point out of a set of detected Shi-Tomasi points, then performs blanket spreading until the blanket is considered to be spread based on wrinkle detection. In another study [16], rather than relying only on camera data during sheet spreading, the authors also used sensors attached to their robot's specially-designed grippers.

In general, learning-based methods for robotic bed-making employ deep neural networks to extract features from image data of the bed scenario (e.g. pillow, blanket configuration) and/or robot state (e.g. joint angles, end-effector position) [17–19]. More specifically, in [17], two separate neural networks were jointly trained using imitation learning and were used to perform blanket spreading. The first network is the grasp policy, which was used to detect blanket corners; the second network is the transition policy, which was used to decide at a given time whether or not to move to the other side of the bed in order to spread the sheet on that side. The system was able to successfully spread the sheets over the bed in most cases, even when additional items (e.g. toys) were present on the surface. This work was extended by [18], which incorporated depth images into the robot's perception of the environment, treated sheet corners as grasp points, and used a more diverse set of initial sheet states. While [17, 18] explored the blanket spreading task as a whole, [19] focused on improving the choice of pick points on the fabric. The authors applied deep transfer learning to train a neural network to select pick-up points on which a robot can act to effectively smooth out blankets.

Unfortunately, despite their titles, the above robotic bed-making studies only focus on sheet spreading, thus, they cannot be directly applied to solve the pillow placement problem which has its own set of technical challenges and cannot be neglected during the development of a bed-making robot.

2.2 Environment Understanding

In order to place the pillows, the robot must be able to gather position and orientation information about the pillows on the bed. With the image feed captured from the robot's perspective, the robot has two objectives: the first is to locate the pillows by framing the pillows in the image with bounding boxes; the second is to estimate the pillow orientation. These two objectives are jointly referred to in this work as pillow detection and orientation estimation (pillow DOE).

Although the literature on pillow DOE is non-existent, inspiration can be taken from research on the DOE of other objects. For example, DOE methods have been used for DOE of objects (e.g. cars, household objects) from the ground view [20, 21] and objects (e.g. cars, buildings) from aerial and remote sensing views [22–25]. For the former, studies have used methods such as support vector machines [20] and convolutional neural networks (CNN) [21]. Unfortunately, these studies [20, 21] are limited to images in which there is only one object of interest and that object is at the centre of the image. Because pillows are not necessarily centred on the bed image, nor is there necessarily only one pillow, these studies are not directly applicable to pillow DOE.

Methods for DOE in aerial and remote sensing applications [22–25] do not make these assumptions since multiple objects must be analyzed in those images. For example, building DOE involves identifying and analyzing buildings in overhead images. Because buildings can be of any orientation, methods that use axis-aligned bounding boxes (e.g. object detection methods such as R-CNN [31]) are not sufficient. For building DOE, oriented bounding boxes are required to get tighter and more precise bounding boxes around the buildings. One approach is to modify a region proposal network (RPN) [32] to be able to not only propose axis-oriented regions but also angled ones as well [22]. The need for oriented bounding boxes extends beyond the DOE for buildings. To detect planes, boats, and vehicles in remote sensing images, [23] used a combination of FPN [33] and RPN. To detect vehicles in aerial images, [24] performed modifications on SSD [34]. Other methods use separate models for detection and orientation estimation [25]. Unfortunately, these studies are limited to images where the objects are visible to the camera's point-of-view and are not obscured by other objects (e.g. clouds).

In contrast, the bed-making scenario would include clutter and other objects (e.g. blankets, toys) that would affect the robot's ability to sense the pillows. In addition, pillows can exhibit many different attributes and be located on beds of different characteristics. For example, to increase user comfort, pillows are designed using different materials [35] and shapes [36]. As a result, methods for pillow DOE not only have to accurately provide position and orientation information about the pillow, but also account for the high variability in pillow appearance. The above studies focus on objects other than pillows and rely on assumptions (e.g. one object per image, object centered in image, objects unobscured in image) that are unreasonable for bed-making, and thus are not directly applicable to robotic pillow placement.

2.3 Manipulation

In pillow placement, object manipulation is another key aspect to consider. A common assumption made in many robotic object manipulation studies is that the object is rigid [30]. However, many objects with which we interact every day in living environments, such as pillows, are deformable. Deformable object manipulation is challenging for various reasons: 1) There is not a trivial approach to representing the configuration of non-rigid objects [37], and 2) the dynamics of deformable objects are complex and non-linear [38]. These reasons present challenges when trying to model the object during task and motion planning. Various studies have attempted to address these challenges using techniques such as classical control [26], robust control [27], and adaptive control [28]. Reinforcement learning (RL) has also been explored, with some methods requiring expert demonstrations [29] while others relying on the robot's own exploration [30]. However, the RL process is highly data- and computation-intensive and is not feasible for low-resource development and operation situations. Simulation can enable RL models to be trained more efficiently due to simulations 1) being quicker because of the lack of physical hardware limitations in the simulation world and the ability to increase real-time factor, and 2) being able to generate a larger, more diverse amount of training data because simulation designers can vary environment parameters and create different environments [39]. However, discrepancies between simulation and real-world makes it challenging to apply simulation-trained robotic behaviour to the real world [40]. This is especially the case for the living environment, where each setting is unique. Furthermore, the pillows themselves can also be unique as well, as they can vary in shape, weight, material, dimensions, and deformation properties. This physical variability results in the above existing approaches for manipulation [26–30] not being able to be directly applied to the pillow placement problem, thus motivating the need to develop solutions for pillow manipulation.

3. PROPOSED PILLOW PLACEMENT METHODOLOGY

In this section, we present the proposed pillow placement methodology for enabling a 6-DOF one-armed robotic manipulator to autonomously place pillows at desired goal poses on a bed (Fig. 2). The main idea of the approach is to first locate the pillows using an onboard camera, and then manipulate the pillows according to the pillows' original poses and goal poses. More specifically, the proposed architecture is as follows: 1) The over-the-shoulder camera gathers RGB-D information about the bed; 2) The pillow detection and orientation estimation process (Pillow DOE Process) uses the robot view to estimate the position and orientation of the pillows on the bed (Sec. 3.2); 3) If there are still pillows to be placed, one of the pillows and its corresponding goal are chosen based on Euclidean-distance-based comparison (Sec. 3.3); and 4) The robot actuates its arm and moves the chosen pillow using macro-actions that are based on pillow position relative to the robot (Sec. 3.4). The remainder of this section will discuss each module in further detail.

3.1 Conventions

In this research, five reference frames were used (Fig. 3): the robot frame $\mathcal{F}_R$ is attached to the robot's base, and is set equal to the global frame $\mathcal{F}_G$ because the robot base is static; the camera frame $\mathcal{F}_S$ is attached to the robot's over-the-shoulder camera; the overhead frame $\mathcal{F}_O$ is attached to an unused overhead camera that provides a bird's-eye-view of the bed setup; and the image frames $\mathcal{F}_{SI}$ and $\mathcal{F}_{OI}$ are the frames of the image captured by the robot camera and overhead camera respectively. For simplicity and without loss of generality, the axes of $\mathcal{F}_{OI}$ were set to be parallel to the global frame, and the bed's surface was set to be coplanar to the global x-y plane.

The pillow's position is the position of the pillow's center. The pillow's orientation is defined to be the angle of the longest edge of the pillow and is taken with respect to the horizontal axis and about the axis going into the page. This angle is limited to be within the range $[-\frac{\pi}{2}, +\frac{\pi}{2}]$.

3.2 Pillow DOE Process

The goal of the Pillow DOE Process is to provide pose information about the pillows in order for manipulation to occur. Specifically, the Pillow DOE Process returns the pose $p_{OI}^{(i)} := \begin{bmatrix} u^{(i)} & v^{(i)} & \alpha^{(i)} \end{bmatrix}^T$ of the i^{th} pillow with respect to $\mathcal{F}_{OI}$, where $u^{(i)}$ and $v^{(i)}$ denote the 2D position and $\alpha^{(i)}$ is the orientation. The process comprises two main steps: First, the RGB image from the camera (I_{SI-RGB}) is transformed from $\mathcal{F}_{SI}$ into $\mathcal{F}_{OI}$; Second, the transformed image is fed into the Pillow DOE method, which consists of a detection stage and an orientation estimation stage that infers position and orientation of the pillows in the transformed image. The rest of Sec. 3.2 provides further details.

R2O Transformation. In this stage, the robot view is transformed into the Robot-to-Overhead (R2O) View (Fig. 4 middle column). The goal of the R2O transformation is to use the robot view to generate an image (i.e. the R2O image) that looks as if the image was taken from the $\mathcal{F}_O$. This conversion is done because the robot in this research (Sec. 4.1) does not use an overhead camera. The conversion involves using a projective transformation [41] to map every pixel of the robot view to the overhead frame [15]. Specifically, each pixel in the robot view RGB image I_{SI-RGB} is transformed based on 1) the pose of the robot camera with respect to the $\mathcal{F}_O$ and 2) the depth value corresponding to that pixel, obtained from the robot view depth image I_{SI-D}. The output is a set of images (RGB and depth) that resembles what the environment would look like from the $\mathcal{F}_O$, specifically

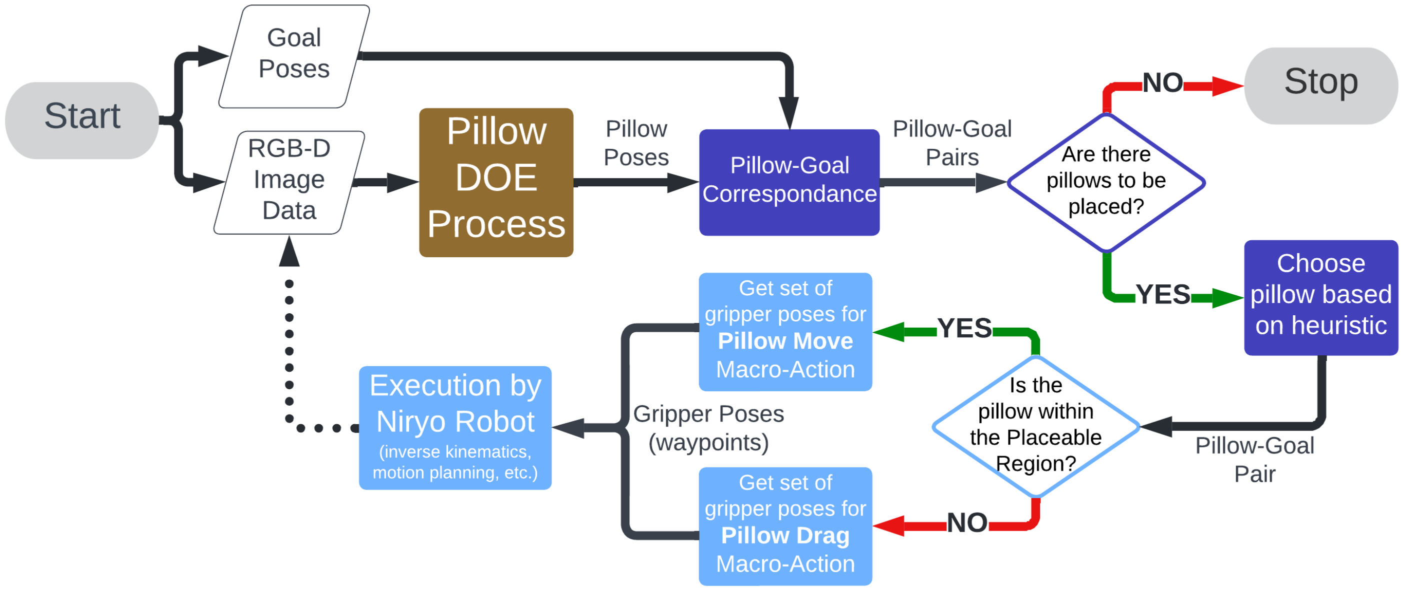

FIGURE 2: PROPOSED PILLOW PLACEMENT PROCESS. THE THREE MAIN STAGES ARE PILLOW DOE PROCESS (IN BROWN, SEC. 3.2), PILLOW MANIPULATION PLANNING (IN DARK BLUE, SEC. 3.3), AND PILLOW MANIPULATION EXECUTION (IN LIGHT BLUE, SEC. 3.4).

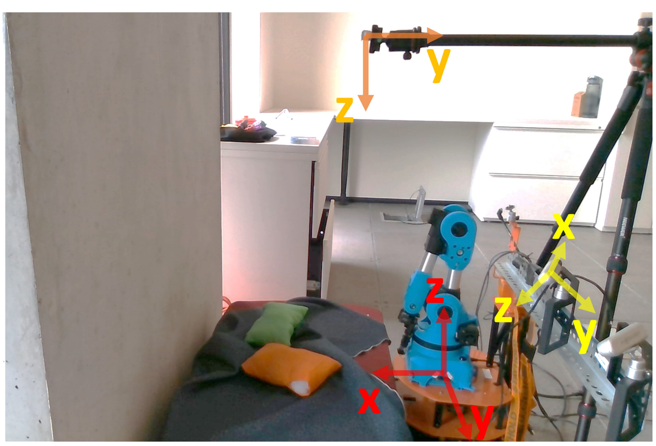

FIGURE 3: PHYSICAL REFERENCE FRAMES USED IN THIS RESEARCH. SUPERIMPOSED ON THE IMAGE OF THE BED SETUP ARE $\mathcal{F}_R = \mathcal{F}_G$ (IN RED), $\mathcal{F}_S$ (IN YELLOW), AND $\mathcal{F}_O$ (IN ORANGE).

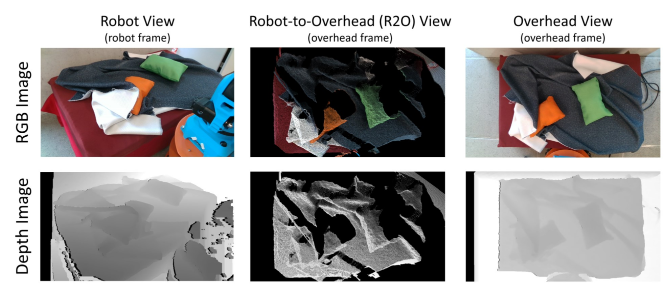

FIGURE 4: DIFFERENT VIEWS AND THE REFERENCE FRAME TO WHICH EACH BELONGS. THE IMAGES FROM LEFT TO RIGHT ARE AS FOLLOWS: TOP ROW SHOWS I_{SI-RGB}, I_{R2O}, AND I_{OI-RGB}; BOTTOM ROW SHOWS I_{SI-D}, I_{R2O-D}, AND I_{OI-D}.

$I_{R2O-RGB} = I_{R2O}$ and I_{R2O-D} respectively. These R2O images, which would be similar to their overhead view counterparts I_{OI-RGB} and I_{OI-D}, are useful for cases where an overhead view is required but the hardware to provide that view is not available.

Detection. Pillow detection was performed on I_{R2O} using a trained YOLOv4-tiny model [42]. YOLOv4-tiny was chosen because it is a lighter version of YOLOv4 [43], which has shown to provide accurate detections in real-time applications (e.g. [44]). Furthermore, the lighter version decreases the amount of computation and memory that the model would require onboard [45]. The output is a set of bounding boxes that frame the pillows in the image. The pixel coordinates of the centre of each bounding box are taken to be the position (u, v) of the corresponding pillow. YOLOv4-tiny is also capable of providing confidence values for each detection; this value is crucial in determining which detections are kept and which are ignored. The YOLOv4-tiny model used in the experiments was trained via transfer learning where a pre-trained YOLOv4-tiny model (trained on the MS COCO dataset [46]) was trained on labelled I_{R2O} of different bed-making scenarios. I_{SI-RGB} and I_{SI-D} were captured from various locations around the bed and not solely from the robot view used in the setup. Both quantity and pillow pose were varied to increase the diversity of the dataset. The custom dataset consisted of 7,700 images, and the training-validation-testing split was 0.85:0.1:0.05.

Orientation Estimation. In this stage, the bounding box areas outputted by the detection model are first extracted from I_{R2O}, producing smaller images I_P of each pillow. Then, each I_P is fed into OE-CV, a multi-stage process that analyzes I_P using classical computer vision methods to estimate pillow orientation α. The stages are visualized in Fig. 5. Pillow colour (image

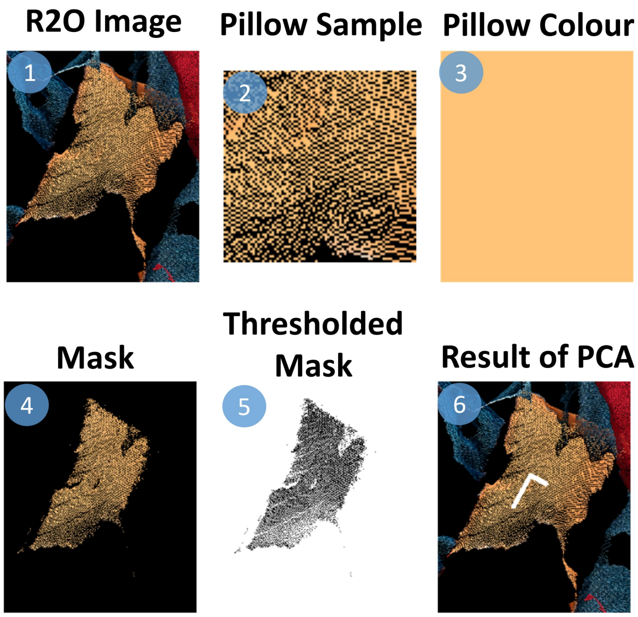

FIGURE 5: ORIENTATION ESTIMATION USING CLASSICAL COMPUTER VISION (OE-CV)

3) is first inferred from a sample (image 2) of the centre of the R2O pillow region (image 1) given by step 3 of the Pillow DOE Process. Colours that are dissimilar to the pillow colour are masked (image 4) and thresholded out to isolate the pixels containing the pillow (image 5). The first principal component (obtained via PCA [47]) of the remaining pixels provides the direction of greatest variance and is used to estimate α. Image 6 shows the first and second principal components as the longer and shorter white lines respectively. α is calculated using the angle of the first principal component, which completes the process of determining p_{OI}.

3.3 Pillow Manipulation Planning

In this stage, manipulation actions are planned based on the poses p_{OI} gathered from the Pillow DOE Process. p_{OI} are first transformed from $\mathcal{F}_{OI}$ into $\mathcal{F}_G$ using a static transformation matrix determined prior to execution by collecting 3D-2D point correspondences and solving a PnP-like problem. Following this, each pillow is assigned a goal pose $p_g^{(i)}$ in the global frame to which the robot will move the pillow. The pillow-goal correspondence problem is formalized as follows: Given a set of m_g goal poses and m_c current pillow poses, where $m_g \geq m_c$, determine for each pillow which p_g to which they should be moved. The approach used in this paper to assign a goal pose to each pillow was to determine the pair that minimized the total Euclidean distance between pillows and goals; the Euclidean distance provides the path of shortest distance along which the pillow can travel to reach its desired pose. Let $p_c^{(i)}$ be the position of the i^{th} pillow in the global frame, where $i = 1, 2 \ldots m_c$; $g(\cdot)$ be the correspondence function that maps from current positions to corresponding goal positions $p_g^{(i)}$; and $D(\cdot,\cdot)$ be the Euclidean distance function that computes the distance between two positions. Then the correspondence function g^*, which provides the pillow-goal correspondences, is

$$g^* = \arg\min_g \sum_{i=0}^{m_c} D\left(p_c^{(i)}, g(p_c^{(i)})\right) \tag{1}$$

Since this algorithm has an $O\left(\frac{m_g!}{(m_g - m_c)!}\right)$ time and space complexity, it quickly becomes computationally infeasible as the number of goals and pillows increases. However, for a sufficiently small number of goals and pillows, such as the $m_c \leq 2$ in this experiment, this simple algorithm is acceptable.

3.4 Pillow Manipulation Execution

The goal of this stage is to move the pillows to their desired goal poses. Pillow manipulation execution can be decomposed into two main steps: a coarse adjustment step and a fine adjustment step. In the coarse adjustment step, the pillow is moved from the start pose to a pose that is near the p_g, specifically within arbitrary radius R from p_g where R is usually the maximum deviation between p_g and current pose after the coarse adjustment step. In the fine adjustment step, the pillow is adjusted to p_g.

Based on the pillow-goal pairs $\{(p_c^{(i)}, p_g^{(i)}) | i = 1, 2, \ldots, m_c\}$ from the manipulation planning stage, a set of gripper poses are calculated as position waypoints $r^{(j)}$ for the robot to follow, where j is the waypoint index. The specific set of waypoints used $\{r^{(j)} | j \in \mathbb{Z}^+\}$ depends on which action is executed. The robot uses two distinct macro-actions to manipulate the pillows: Pillow Move and Pillow Drag, both of which include a series of primitive actions (i.e. joint movements) performed by the robot when given a set of gripper poses.

The robot is only expected to be able to place pillows that it can access with its gripper. The area of the bed which a robot can reach is referred to in this work as the robot's Accessible Region $\mathbb{R}_A$. For static manipulators that are constrained by their dimensions and joint configuration, such as that used in the experiments, the $\mathbb{R}_A$ is constant and limited to a certain portion of the bed (Fig. 6 right). The robot manipulates pillows in $\mathbb{R}_A$ using the two different macro-actions.

Pillow Move Macro-Action. The Pillow Move macro-action moves the pillow from the p_c to p_g. The sequence of primitive actions involves positioning the gripper above the pillow centre, grasping the pillow from above, then moving the pillow to p_g. By positioning the gripper above the pillow centre, the pillow's position on the bed can be controlled by specifying the gripper's position above the bed. For example, if $p_g = (x_{goal}, y_{goal}, h)$ where h is the height of the pillow, then the goal gripper location that is required to move the pillow would be $(x_{goal}, y_{goal}, h + \Delta)$ where Δ is the distance above the pillow that the gripper must be to grasp the pillow. By grasping from above, the orientation can be changed through pronation and supination of the gripper wrist (joint 6 in Fig. 1). Because this approach entails the gripper to be pointing down instead of away from the robot, the region that the robot can access using the Pillow Move action is only a portion of the total Accessible Region: This smaller region is referred to in this research as the Placeable Region $\mathbb{R}_P$ where $\mathbb{R}_P \subseteq \mathbb{R}_A$.

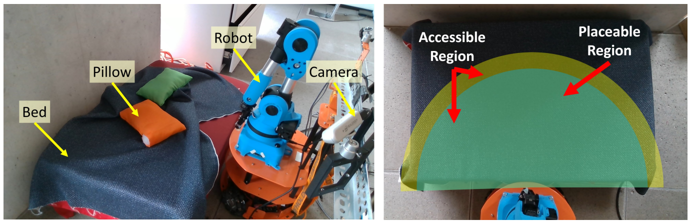

FIGURE 6: BED-MAKING SETUP WITH THE MAJOR COMPONENTS LABELLED (LEFT). ROBOT ACCESSIBLE REGION IS SHOWN IN GREEN AND YELLOW WHILE THE PLACEABLE REGION IS SHOWN IN GREEN (RIGHT).

This is thus the region in which the pillow centre must be located in order for the Pillow Move action to be possible (Fig. 6 right).

Pillow Drag Macro-Action. The Pillow Drag macro-action is performed when the pillow's centre falls outside of $\mathbb{R}_P$ (but still within $\mathbb{R}_A$). In this case, the robot can no longer reach the pillow centre from above and has to reach the pillow from the side. The goal of the Pillow Drag action is to drag the pillow closer to the robot and into $\mathbb{R}_P$ so that the Pillow Move action could be performed. The Pillow Drag action is an important capability for the robot to have for future developments because pillows can easily be outside of $\mathbb{R}_P$ during real-life bed-making situations.

4. REAL-WORLD EXPERIMENTS

4.1 Setup

Environment Setup. This research used a miniature bed-making scenario of 2:5 scale (Fig. 6 left). The robot was placed near the bed, halfway along the longer side of the bed. A mock bed was created to model an actual-sized bed. The bed is 0.50 m x 0.75 m, and consists of a soft bottom portion (in red) and a blanket that covers that portion (in dark blue and white). Two pillows were used in the experiments. The orange pillow is 13 cm x 20 cm x 6.5 cm while the green pillow is 13 cm x 20 cm x 7.5 cm. Each pillow comprises a cotton filling enclosed within a polyester pillowcase.

Robot Setup. The robot used in the experiments was the Niryo One robot [48], a 6-DOF jointed arm robotic manipulator with a 2-finger gripper as its end effector (Fig. 1). Mounted in an over-the-shoulder position on the left-hand side of the robot is an Intel RealSense Depth Camera D415 sensor [49] that provides the system with 1280 x 720 pixel RGB-D images of the environment. The robot's six DOFs allow it to have a robot workspace that is sufficient for accessing >50% of the bed plane $\mathbb{R}_B$. However, due to the robot's size, the robot's reach at a given base position is limited to a certain area of the bed, hence $\mathbb{R}_A < \mathbb{R}_B$.

Experiment Setup. Experiments were conducted to test the proposed bed-making process in terms of 1) the performance of the Pillow DOE Process in estimating position and orientation of pillows and 2) the success rate in placing pillows. The former was measured using three metrics during the Pillow DOE experiments: 1) Intersection over Union (IoU); 2) Bounding Box Deviation (BBD), which is the distance between the centre of the ground truth bounding box and that of the predicted bounding box; and 3) Percentage of Predictions within specific ranges, which is denoted as PP@ρ where ρ is the angle range in degrees. The metric used to measure overall pillow placement performance during the Pillow Placement experiments was the percentage of pillows successfully placed by the robot. This metric was chosen because it directly quantifies the methodology's ability to enable robotic pillow placement.

4.2 Pillow DOE Experiments

Detection and orientation estimation were tested separately first before testing them together. The purpose of this experiment was to evaluate the DOE method's ability to perform pillow DOE and what weaknesses should be considered. The 385 images used in this experiment were taken from the test set.

In the detection test, the trained YOLOv4-tiny model was tested on its ability to accurately surround pillows in I_{R2O} with bounding boxes. This test was conducted by first resizing I_{R2O} to the 416 x 416 pixel images required by YOLOv4-tiny and then using the resized images as input to the model. In the orientation estimation test, the OE-CV model was tested on its ability to accurately assign orientations to pillows. To test this, I_P were generated by cropping image sections containing pillows from the original I_{R2O} and inputted into the OE-CV model. In the full DOE test, the detection and orientation estimation capabilities were combined and tested on their ability to accurately determine the poses of pillows in the image.

4.3 Pillow Placement Experiments

In the pillow placement experiment, the full pillow placement process was tested. The $\mathbb{R}_A$ and $\mathbb{R}_P$ were identified prior to the experiments. During the experiment, a set of runs were performed, where each run consisted of the following steps: 1) Randomly generate start poses $\{p_c^{(i)} \mid i = 1, 2, \ldots, m_c\}$ and goal poses $\{p_g^{(i)} \mid i = 1, 2, \ldots, m_g\}$ for the pillows subject to the constraints imposed by $\mathbb{R}_A$ and $\mathbb{R}_P$. Specifically, $p_c^{(i)} \in \mathbb{R}_A, p_g^{(i)} \in \mathbb{R}_P\ \forall i$; 2) Manually place the pillows at $p_c^{(i)}$; 3) Run the pillow placement process and allow the robot to autonomously place the pillows. A run was successful if the final poses of the pillows reached $p_g^{(i)}$ within acceptable error $\epsilon = (\epsilon_{x,y}, \epsilon_\theta)$. In this experiment, 50 runs were performed, and $\epsilon = (3.7\text{ cm}, 15°)$.

5. RESULTS AND DISCUSSION

In this section, we discuss the results for each of the experiments that were mentioned in the previous section, namely those for pillow DOE and pillow placement.

5.1 Pillow DOE Experiments

Detection Experiment. Table 1 presents a summary of the detection performance, which shows that the positions of the pillow centres outputted by the detection model are only on average 2.4 cm away from the ground-truth position. This level of error is acceptable because the robot could place pillows successfully during the experiments despite small errors (<5 cm deviation).

Since one characteristic of an unmade bed is its messiness, we also explored the effect of introducing folds in the blanket and objects of varying dimensions and colours onto the bed. A constant number of objects were used, and both the folds and

TABLE 1: PERFORMANCE OF YOLOV4-TINY ON DETECTING PILLOWS

Metric	Mean	Min	Max
Intersection over Union (IoU)	75.9%	51.4%	93.4%
Bounding Box Deviation (BBD)	2.4 cm	0.55 cm	5.4 cm

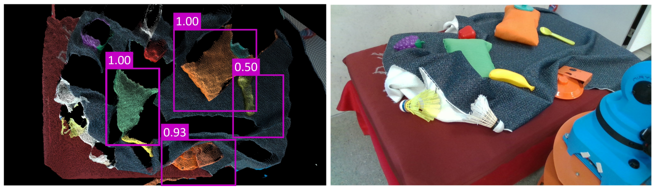

FIGURE 7: EXAMPLE OF USING THE DETECTION MODEL FOR A CLUTTERED BED. THE LEFT-HAND IMAGE SHOWS THE DETECTION RESULTS; MAGENTA BOUNDING BOXES INDICATE THE DETECTIONS, AND THE CORRESPONDING CONFIDENCES ARE INCLUDED ABOVE THE BOUNDING BOXES. THE RIGHT-HAND IMAGE SHOWS THE CORRESPONDING ROBOT VIEW.

objects partially covered the pillows, an aspect not encountered in the dataset. For example, Fig. 7 shows an example where the green pillow and orange pillow were only approximately 80% visible due to the blanket and objects. In this example, although both pillows were not fully visible, the model successfully detected both pillows. Furthermore, the model was able to ignore other objects on the bed (e.g. the plastic toys of varying colors). For the non-pillow objects detected, detections with <60% confidence (e.g. the green-yellow spoon near the right-hand side of the image) were ignored by setting the confidence threshold to be ≥ 60%. However, the orange non-pillow object at the bottom was detected with 93% confidence, and hence would be treated as a valid detection if the threshold was set to be less than 93%.

Orientation Estimation Experiment. Figure 8 shows an example of the results. A common occurrence for errors was when the R2O version of pillows was shaped as right-angled triangles (Fig. 8 bottom-right). In these cases, the direction of the principal component found using PCA during the OE-CV process was not in the direction of the desired orientation line, but rather in the direction of the triangle's hypotenuse. Despite this weakness, OE-CV was accurate within 30° for 98% of the tests (Table 2).

DOE Experiment. Figure 9 shows an example of the results. As expected from Tables I and II, the predicted bounding boxes and orientation lines in the DOE experiment were very similar to the ground truth: BBD<6 cm and PP@30=95% . The last row shows an interesting case. The green pillow is detected as having its long side closer to being parallel to the image x-axis than to being parallel to the y-axis (i.e. the pillow seems to be horizontal). However, in reality, the pillow is closer to being vertical. The robot view shows the reason for this error: The green pillow was not fully visible to the camera (i.e. <20% of the possible visible area), and the DOE method inferred that the pillow was more horizontal than vertical based on the limited information. In the pillow placement experiments, this error in the pillow

TABLE 2: PERCENTAGE OF OE-CV PREDICTIONS FALLING WITHIN SELECT RANGES

Metric	OE-CV Performance
10° from the truth (PP@10)	53%
20° from the truth (PP@20)	80%
30° from the truth (PP@30)	98%

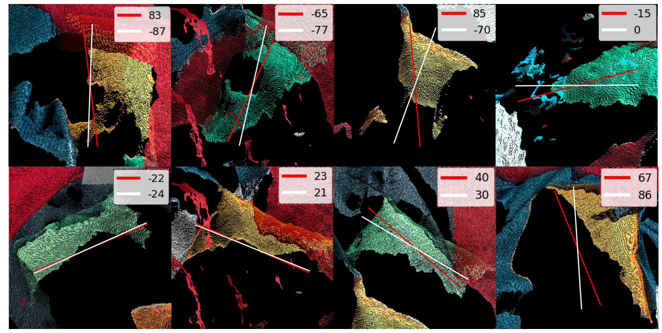

FIGURE 8: OE RESULTS SHOWING VARYING OE-CV PERFORMANCES. ORIENTATION LINES SUPERIMPOSED ON THE R2O IMAGES INDICATE PILLOW ORIENTATION. THE RED AND WHITE ORIENTATION LINES ARE THE OE-CV OUTPUT AND GROUND TRUTH RESPECTIVELY. THE ANGLES IN THE LEGENDS ARE IN DEGREES.

orientation was not a concern since the pillows were assumed to be sufficiently visible (>50% of the possible visible area) for the robot to detect the pillows. For cases where the deviation between the inferred pose and the true pose is sufficiently large (>7 cm), the robot would fail to grasp the pillow and would try again in the next iteration.

5.2 Pillow Placement Experiments

A video of one of the pillow placement runs is presented at https://youtu.be/BXcwffPKcGI. Table 3 shows a summary of the results. Pillow placement was successful in 89% of the runs. One of the common issues encountered during the runs was when pose adjustment was small (<15° or <1 cm from the pillow's undeformed state). In those cases, the robot's force would only contribute to deforming a portion of the pillow and not change the pillow's pose. As a result, actions that performed small pose adjustments were only successful approximately 50% of the cases where the required pose adjustment was <15° or <1 cm. For example, during small orientation adjustments, the gripper only deformed the pillowcase while the rest of the pillow remained stationary.

TABLE 3: PILLOW PLACEMENT EXPERIMENTAL RESULTS AT A GLANCE

	Correct Position	Incorrect Position
Correct Orientation	89%	1%
Incorrect Orientation	3%	7%

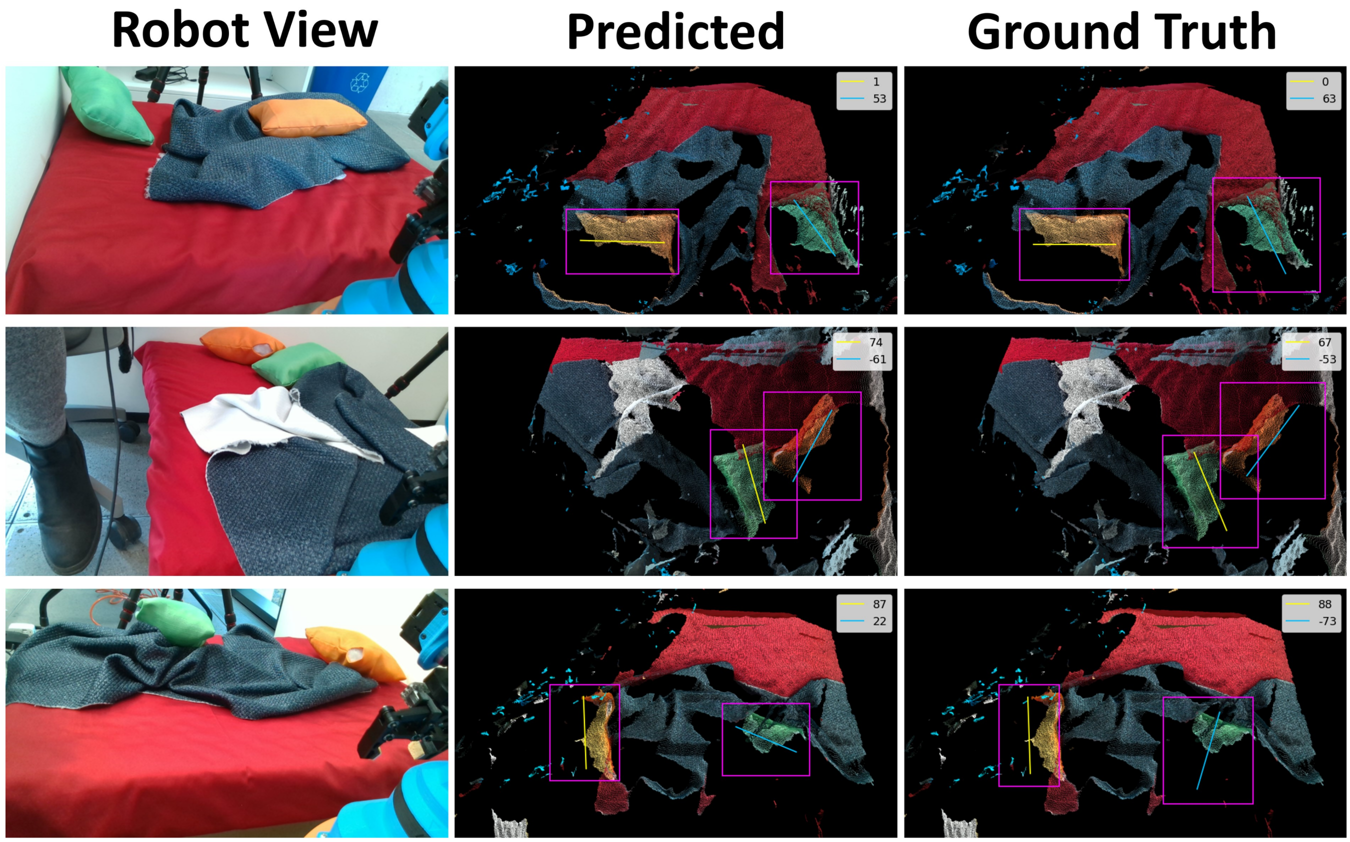

FIGURE 9: DOE RESULTS. THE ROWS SHOW RESULTS OF PERFORMING PILLOW DOE ON VARIOUS ROBOT VIEWS. THE FIRST COLUMN SHOWS THE ROBOT VIEW. THE SECOND COLUMN SHOWS THE BOUNDING BOXES (IN MAGENTA) AND ORIENTATION LINES (IN YELLOW AND CYAN) (WHOSE ANGLE IS THE PILLOW ORIENTATION) SUPERIMPOSED ON THE R2O IMAGE. THE THIRD COLUMN SHOWS THE GROUND TRUTH DOE. THE ANGLES IN THE LEGENDS ARE IN DEGREES.

The iterative nature of the process allowed the robot to attempt the manipulation again in the following iteration if the manipulation was unsuccessful in the current iteration. The number of actions performed in each run varied based on whether or not the pillows were within $\mathbb{R}_P$. Another factor that increased the number of actions required was the number of imperfect places: Every time the robot moved the pillow to the goal pose and the pillow did not reach the goal pose, the robot would have to perform the action again in the following iteration. The mean number of actions per pillow placed s_{mean} was calculated as follows:

$$s_{mean} = \frac{1}{n_P} \sum_{k=0}^{n_{runs}} \sum_{i=0}^{m_c} s_k^{(i)} \tag{2}$$

where n_P is the total number of pillows placed, n_{runs} is the number of runs, and $s_k^{(i)}$ is the number of actions required to place the i^{th} pillow during the k^{th} run. Across the n_P =50 runs, s_{mean} =2.8 while the minimum possible mean number of actions needed was 1.3. This minimum value was calculated based on the minimum number of actions required to place pillows within $\mathbb{R}_P$ being 1 (i.e. one Pillow Move action) and that for pillows outside of $\mathbb{R}_P$ being 2 (i.e. one Pillow Drag action and one Pillow Move action). The experimental mean is greater than the minimum possible mean, which agrees with the observation that pillows were often not placed perfectly the first time the Pillow Move action was performed.

6. CONCLUSION

In this paper, we presented a novel process that enables a static one-armed 6-DOF robotic manipulator to address the challenges of robotic pillow placement. While existing literature focuses on the sheet spreading part of the bed-making task, the robotic pillow placement task has been neglected despite its unique challenges. To fill this gap, our approach included the use of learning-based object detection, classical computer vision methods, and manipulation macro-actions, all arranged in a sequential pillow placement process. We evaluated the proposed research in terms of the percentage of pillows successfully placed, where results showed the feasibility of the proposed process in enabling robotic pillow placement. While the robot was able to accomplish the pillow placement task for 89% of the runs, limitations include the following: 1) the dataset used to train the DOE method was not representative of bed scenarios a bed-making robot would encounter in real life; 2) the diversity of pillows used is limited. Thus, future work is needed test the robot in more bed-making scenarios in order to enable the system to be more robust to changes in the environment.

ACKNOWLEDGMENTS

The authors would like to acknowledge the other members of the Robotics and Automation Laboratory (RAL) at the University of Toronto for their support in constructing the bed-making image dataset that was instrumental to this research.

REFERENCES

[1] Lawton, M. Powell. "Aging and Performance of Home Tasks." *Human Factors* Vol. 32 No. 5 (1990): pp. 527–536. DOI 10.1177/001872089003200503. PMID: 2074107.

[2] Gillick, Muriel R. "The critical role of caregivers in achieving patient-centered care." *JAMA* Vol. 310 No. 6 (2013): p. 575. DOI 10.1001/jama.2013.7310.

[3] Abdelmoneium, Azza, Rankin, Janet and Corman, Michael. "Roles and Responsibilities of Family Home Caregivers for Elderly People in Qatar: Experiences and Challenges." *International Journal of Social Science Studies* Vol. 5 No. 7 (2017): pp. 1–11. DOI 10.11114/ijsss.v5i7.2413.

[4] Foner, Nancy. *The caregiving dilemma work in an American Nursing Home*. Univ. of California Press (1995).

[5] "Preventing and managing COVID-19 across long-term care services." Policy brief. World Health Organization. 2020.

[6] Kangasniemi, Mari, Karki, Suyen, Colley, Noriyo and Voutilainen, Ari. "The use of robots and other automated devices in nurses' work: An integrative review." *International Journal of Nursing Practice* Vol. 25 No. 4 (2019): p. e12739. DOI https://doi.org/10.1111/ijn.12739.

[7] Beer, Jenay, Smarr, Cory-Ann, Chen, Tiffany, Mitzner, Tracy, Kemp, Charles and Rogers, Wendy. "The Domesticated Robot: Design Guidelines for Assisting Older Adults to Age in Place." Vol. 2012. 2012. DOI 10.1145/2157689.2157806.

[8] Ezer, Neta, Fisk, Arthur D. and Rogers, Wendy A. "More than a Servant: Self-Reported Willingness of Younger and Older Adults to having a Robot perform Interactive and Critical Tasks in the Home." *Proceedings of the Human Factors and Ergonomics Society Annual Meeting* Vol. 53 (2009): pp. 136 – 140.

[9] Bloomfield, Jacqueline, Pegram, Anne and Jones, Anne. "Recommended procedure for bedmaking in hospital." *Nursing Standard* Vol. 22 No. 23 (2008): p. 41–44. DOI 10.7748/ns2008.02.22.23.41.c6419.

[10] Freitag, Sonja, Ellegast, Rolf, Dulon, Madeleine and Nienhaus, Albert. "Quantitative Measurement of Stressful Trunk Postures in Nursing Professions." *The Annals of Occupational Hygiene* Vol. 51 No. 4 (2007): pp. 385–395. DOI 10.1093/annhyg/mem018.

[11] Milburn, Peter and Barrett, Rod S. "Lumbosacral loads in bedmaking." *Applied Ergonomics* Vol. 30 No. 3 (1999): pp. 263–273. DOI https://doi.org/10.1016/S0003-6870(98)00021-0.

[12] Fausset, Cara Bailey, Kelly, Andrew J., Rogers, Wendy A. and Fisk, Arthur D. "Challenges to Aging in Place: Understanding Home Maintenance Difficulties." *Journal of Housing For the Elderly* Vol. 25 No. 2 (2011): pp. 125–141. DOI 10.1080/02763893.2011.571105. PMID: 22072843.

[13] Gitlin, Laura. "Conducting Research on Home Environments: Lessons Learned and New Directions." *The Gerontologist* Vol. 43 (2003): pp. 628–37. DOI 10.1093/geront/43.5.628.

[14] Pellatt, Glynis. "Clinical skills: bed making and patient positioning." *British journal of nursing (Mark Allen Publishing)* Vol. 16 (2007): pp. 302–5. DOI 10.12968/bjon.2007.16.5.23010.

[15] Jennings, Scott, Dong, Qing, Li, Yuanziwei and Goldenberg, Andrew. "Development of an Autonomous Bed-making Robot." (2021). ANVIV Mechatronics Inc. in collaboration with ANVIV Lab at the University of Toronto.

[16] von Drigalski, Felix, Yoshioka, Daiki, Gall, Marcus, Uriguen Eljuri, Pedro, Yamazaki, Wataru, Cho, Sung-Gwi, Hoerig, Viktor, Ullauri, Jessica, Ding, Ming, Takamatsu, Jun and Ogasawara, Tsukasa. "A robotic system for automated bed-making using a gripper specialized for textile manipulation." *The Proceedings of JSME annual Conference on Robotics and Mechatronics (Robomec)* Vol. 2017 (2017): pp. 2P2–F10. DOI 10.1299/jsmermd.2017.2P2-F10.

[17] Laskey, Michael, Powers, Chris, Joshi, Ruta, Poursohi, Arshan and Goldberg, Ken. "Learning Robust Bed Making using Deep Imitation Learning with DART." *ArXiv* Vol. abs/1711.02525 (2017).

[18] Seita, Daniel, Jamali, Nawid, Laskey, Michael, Berenstein, Ron, Tanwani, Ajay Kumar, Baskaran, Prakash, Iba, Soshi, Canny, John F. and Goldberg, Ken. "Robot Bed-Making: Deep Transfer Learning Using Depth Sensing of Deformable Fabric." *ArXiv* Vol. abs/1809.09810 (2018).

[19] Seita, Daniel, Jamali, Nawid, Laskey, Michael, Tanwani, Ajay, Berenstein, Ron, Baskaran, Prakash, Iba, Soshi, Canny, John and Goldberg, Kenneth. *Deep Transfer Learning of Pick Points on Fabric for Robot Bed-Making* (2022): pp. 275–290. DOI 10.1007/978-3-030-95459-8_17.

[20] He, Kun, Sigal, Leonid and Sclaroff, Stan. "Parameterizing Object Detectors in the Continuous Pose Space.": pp. 450–465. 2014. DOI 10.1007/978-3-319-10593-2_30.

[21] Ghosh, Rohan, Mishra, Abhishek, Orchard, Garrick and Thakor, N.v. "Real-time object recognition and orientation estimation using an event-based camera and CNN." *IEEE 2014 Biomedical Circuits and Systems Conference, BioCAS 2014 - Proceedings* (2014): pp. 544–547DOI 10.1109/BioCAS.2014.6981783.

[22] Chen, Yongliang, Gong, Weiguo, Chen, Chaoyue and Li, Weihong. "Learning Orientation-Estimation Convolutional Neural Network for Building Detection in Optical Remote Sensing Image." *2018 Digital Image Computing: Techniques and Applications (DICTA)*: pp. 1–8. 2018. DOI 10.1109/DICTA.2018.8615859.

[23] Yingxue, Chen, Wenrui, Ding, Hongguang, Li, Yufeng, Wang, Shuo, Liu and Xiao, Zhifeng. "Arbitrary-Oriented Dense Object Detection in Remote Sensing Imagery.": pp. 436–440. 2018. DOI 10.1109/ICSESS.2018.8663939.

[24] Tang, Tianyu, Zhou, Shilin, Deng, Zhipeng, Lei, Lin and Zou, Huanxin. "Arbitrary-Oriented Vehicle Detection in

Aerial Imagery with Single Convolutional Neural Networks." *Remote Sensing* Vol. 9 (2017): p. 1170. DOI 10.3390/rs9111170.

[25] Chen, Chaoyue, Gong, Weiguo, Hu, Yongfei, Chen, Yongliang and Ding, Youli. "Learning Oriented Region-based Convolutional Neural Networks for Building Detection in Satellite Remote Sensing Images." *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* Vol. XLII-1/W1 (2017): pp. 461–464. DOI 10.5194/isprs-archives-XLII-1-W1-461-2017.

[26] Sun, Dong, Mills, James K. and Liu, Yunhui. "Position Control of Robot Manipulators Manipulating a Flexible Payload." *The International Journal of Robotics Research* Vol. 18 No. 3 (1999): pp. 319–332. DOI 10.1177/02783649922066231.

[27] Hu, Yan-Ru and Vukovich, George. "Active Robust Shape Control of Flexible Structures." *Mechatronics* Vol. 15 (2005): pp. 807–820. DOI 10.1016/j.mechatronics.2005.02.004.

[28] Navarro-Alarcon, David, Yip, Hiu Man, Wang, Zerui, Liu, Yun-Hui, Zhong, Fangxun, Zhang, Tianxue and Li, Peng. "Automatic 3-D Manipulation of Soft Objects by Robotic Arms With an Adaptive Deformation Model." *IEEE Transactions on Robotics* Vol. 32 (2016): pp. 1–13. DOI 10.1109/TRO.2016.2533639.

[29] Lee, Alex, Gupta, Abhishek, Lu, Henry, Levine, Sergey and Abbeel, Pieter. "Learning from multiple demonstrations using trajectory-aware non-rigid registration with applications to deformable object manipulation.": pp. 5265–5272. 2015. DOI 10.1109/IROS.2015.7354120.

[30] Wu, Yilin, Yan, Wilson, Kurutach, Thanard, Pinto, Lerrel and Abbeel, Pieter. "Learning to Manipulate Deformable Objects without Demonstrations." 2020. DOI 10.15607/RSS.2020.XVI.065.

[31] Girshick, R., Donahue, J., Darrell, T. and Malik, J. "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation." *2014 IEEE Conference on Computer Vision and Pattern Recognition*: pp. 580–587. 2014. DOI 10.1109/CVPR.2014.81.

[32] Ren, Shaoqing, He, Kaiming, Girshick, Ross and Sun, Jian. "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks." *IEEE Transactions on Pattern Analysis and Machine Intelligence* Vol. 39 No. 6 (2017): pp. 1137–1149. DOI 10.1109/TPAMI.2016.2577031.

[33] Lin, Tsung-Yi, Dollár, Piotr, Girshick, Ross, He, Kaiming, Hariharan, Bharath and Belongie, Serge. "Feature Pyramid Networks for Object Detection." *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*: pp. 936–944. 2017. DOI 10.1109/CVPR.2017.106.

[34] Liu, Wei, Anguelov, Dragomir, Erhan, Dumitru, Szegedy, Christian, Reed, Scott, Fu, Cheng-Yang and Berg, Alexander. "SSD: Single Shot MultiBox Detector." Vol. 9905: pp. 21–37. 2016. DOI 10.1007/978-3-319-46448-0_2.

[35] Sung, Min-Jung and Sung, Su-Kwang. "The Effects of Pillow Filling Materials on the Comfortable Sleep." *Fashion & Textile Research Journal* Vol. 8 (2006).

[36] Liu, Shuo-Fang, Lee, Yann-Long and Liang, Jung-Chin. "Shape design of an optimal comfortable pillow based on the analytic hierarchy process method." *Journal of chiropractic medicine* Vol. 10 (2011): pp. 229–239. DOI 10.1016/j.jcm.2011.04.002.

[37] Berenson, Dmitry. "Manipulation of deformable objects without modeling and simulating deformation." *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*: pp. 4525–4532. 2013. DOI 10.1109/IROS.2013.6697007.

[38] Essahbi, Nabil, Bouzgarrou, B. Chedli and Gogu, Grigore. "Soft Material Modeling for Robotic Manipulation." *Applied Mechanics and Materials* Vol. 162 (2012): pp. 184–193. DOI 10.4028/www.scientific.net/AMM.162.184.

[39] Sheckells, Matthew, Garimella, Gowtham, Mishra, Subhransu K. and Kobilarov, Marin. "Using Data-Driven Domain Randomization to Transfer Robust Control Policies to Mobile Robots." *2019 International Conference on Robotics and Automation (ICRA)* (2019): pp. 3224–3230.

[40] Tobin, Josh, Fong, Rachel, Ray, Alex, Schneider, Jonas, Zaremba, Wojciech and Abbeel, Pieter. "Domain randomization for transferring deep neural networks from simulation to the real world." *2017 IEEE/RSJ international conference on intelligent robots and systems (IROS)*: pp. 23–30. 2017. IEEE.

[41] Szeliski, Richard. *Computer vision: Algorithms and applications*. Springer (2010): .

[42] Jiang, Zicong, Zhao, Liquan, Li, Shuaiyang and Jia, Yanfei. "Real-time object detection method based on improved YOLOv4-tiny." (2020)URL 2011.04244.

[43] Bochkovskiy, Alexey, Wang, Chien-Yao and Liao, Hong-Yuan Mark. "YOLOv4: Optimal Speed and Accuracy of Object Detection." *ArXiv* Vol. abs/2004.10934 (2020).

[44] Mahto, Pooja, Garg, Priyamm, Seth, Pranav and Panda, J. "Refining Yolov4 for Vehicle Detection." *International Journal of Advanced Research in Engineering and Technology (IJARET)* Vol. 11 No. 5 (2020): p. 409–419.

[45] Roy, Biparnak, Nandy, Subhadip, Ghosh, Debojit, Dutta, Debarghya, Biswas, Pritam and Das, Tamodip. "Moxa: A deep learning based unmanned approach for real-time monitoring of people wearing medical masks." *Transactions of the Indian National Academy of Engineering* Vol. 5 No. 3 (2020): p. 509–518. DOI 10.1007/s41403-020-00157-z.

[46] Lin, Tsung-Yi, Maire, Michael, Belongie, Serge, Bourdev, Lubomir, Girshick, Ross, Hays, James, Perona, Pietro, Ramanan, Deva, Zitnick, C. Lawrence and Dollár, Piotr. "Microsoft COCO: Common Objects in Context." (2014)DOI 10.48550/ARXIV.1405.0312.

[47] Pearson, Karl. "LIII. On lines and planes of closest fit to systems of points in space." *Philosophical Magazine Series 1* Vol. 2 (1901): pp. 559–572.

[48] "Niryo One, a 6 axis robot arm designed for education and research." Niryo (2022). URL https://niryo.com/fr/product/niryo-one/.

[49] "Intel RealSense Depth Camera D415." Intel (2022). URL https://www.intelrealsense.com/depth-camera-d415/.