RoboMem: Giving Long Term Memory to Robots

Ifrah Idrees
Dept. of Computer Science
Brown University
Providence, USA
ifrah_idrees@brown.edu

Steve P. Reiss
Dept. of Computer Science
Brown University
Providence, USA
spr@cs.brown.edu

Stefanie Tellex
Dept. of Computer Science
Brown University
Providence, USA
stefie10@cs.brown.edu

***Abstract*—Robots have the potential to improve health monitoring outcomes for the elderly by providing doctors, and caregivers with information about the person's behavior, health activities and their surrounding environment. Over the years, less work has been done to enable robots to preserve information for longer periods of time, on the order of months and years of data, and use this contextual information to answer queries. Time complexity to process this massive sensor data in a timely fashion, inability to anticipate the future queries in advance and imprecision involved in the results have been the main impediments in making progress in this area. We make a contribution by introducing RoboMem, a query answering system for health-care assistance of elderly over long term; continuous data feeds that intends to overcome the challenges of giving long term memory to robots. The design for our framework preprocesses the sensor data and stores this preprocessed data into the database. This data is updated in the database by going through successive refinements, improving its accuracy for responding to queries. If data in the database is not enough to answer a query, a small set of relevant frames (also obtained from the database) will be reprocessed to obtain the answer. [Our initial prototype of RoboMem stores 3.5MB of data in the database as compared to 535.8MB of actual video frames and with minimal data in the database it is able to fetch information fundamental to respond to queries in 0.0002 seconds on average].**

***Index Terms*—robot, database, computer vision, memory, query, answering**

I. Introduction

One potential use of robots in the personal assistance of the elderly is to answer questions about their health or their surrounding environment. Such questions can span over days, months or even years of data. Getting these queries answered accurately and timely by robots involves the intersection of many disciplines - robotics, computer vision, databases, natural language processing, and human-computer interaction. Although efforts have been made by Paul et al. [20] to enable robots to respond to commands by incorporating the factual knowledge or observations of its workspace for the last few minutes, this has only yielded a short term memory. We intend to go beyond and provide robots the capability to answer queries over long term memory.

Several challenges exist in building an end-to-end system for giving robots long term memory. These challenges include:

- There is massive sensor data and this information needs to be stored compactly because of the limited storage.
- Initial processing of robot sensor information is not precise enough to answer queries.
- The possible queries are not known in advance therefore we cannot determine what information should be stored to answer them.
- Queries should be responded in a timely fashion, reprocessing all original sensor data cannot be afforded for every question

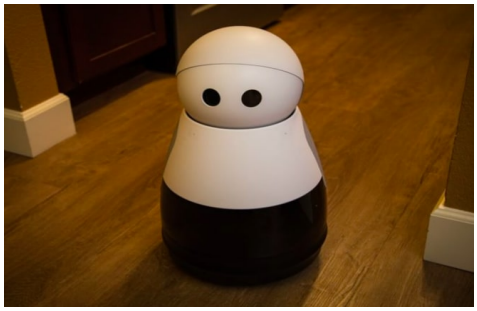

Fig. 1: Kuri Robot [22]

Consequently, we propose a framework; RoboMem, which offers a new approach to robotic memory and perception. RoboMem intends to answer queries that can help assist the care of elderly over periods of days, months, or even years. In particular, we will be providing the social and interactive Kuri robot from Mayfield [17] with a long term memory.

RoboMem intends to overcome the challenges mentioned above with the following design vision:

- Formulating a set of categories for queries which are important for the elderly health-care environment.
- Proposing an architecture which divides data processing into hierarchical phases (pre-processing, post-processing, re-processing).
- Successive refinement of the prepossessed data in the database.
- Answering queries by reprocessing small subset of original sensor data as needed.

Mucchiani et al. [18] conducted a user study highlighting the top 14 activities important in the daily lives of the elderly. We use it as a ground for constraining the set of queries that will be potentially asked from RoboMem. We have cataloged these queries into categories, and these categories along with examples queries are as follows:

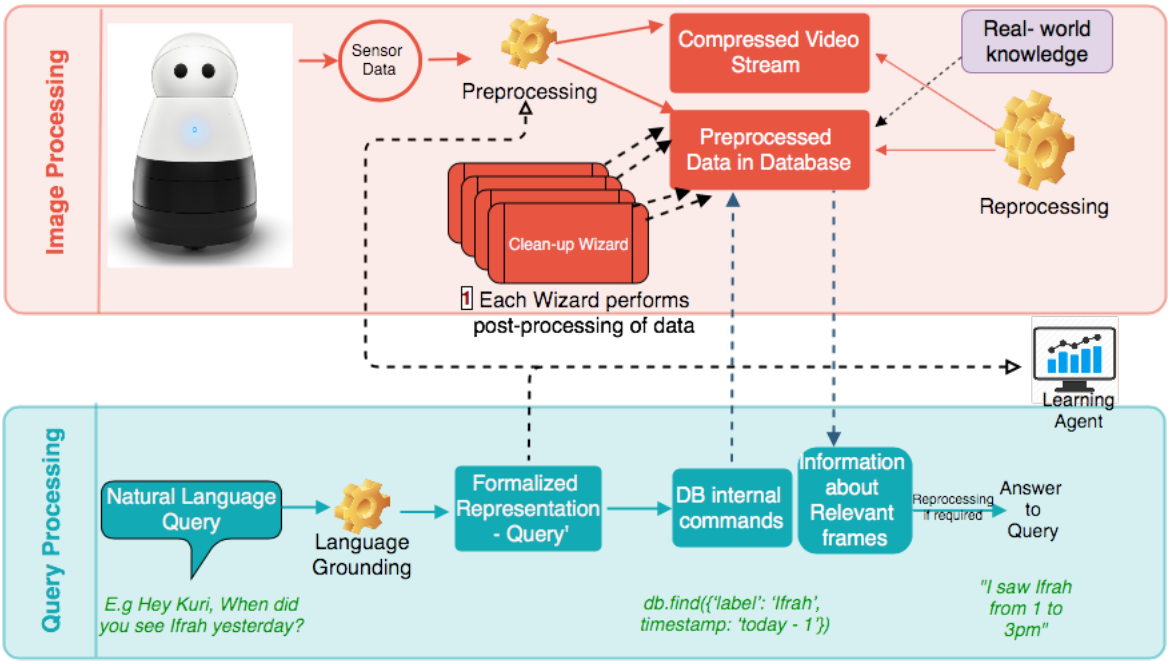

Fig. 2: Workflow Model of RoboMem

1) Spatiotemporal object localization in the surrounding of elderly, e.g. - *"Hey Kuri, do you remember where I last placed my daughter's graduation picture?"*
2) Identifying the people elderly has interacted with e.g. - *"Hey Kuri, can you tell me which of my family members visited me last summer?"*
3) Recognition and spatiotemporal localization of activities performed by elderly e.g.
   - Was the activity performed - *"Hey Kuri, has my dad taken the Montelukast medicine over the past month?"*
   - How long was the activity performed - *"Hey Kuri, how has my patient's sleeping cycle been (how much has my patient slept) over the last three months?"*
   - Where was the activity performed -*"Hey Kuri, where did my patient exercise or walk the most over past week?"*

Input from a physician working with the elderly was used to assist in developing the queries for our initial prototype, and these queries were personally recommended by him since the elderly nearly always have trouble answering these questions by themselves.

The main contributions of our paper are to enable answering questions about elderly's health and their surrounding robustly and efficiently by proposing the *RoboMem* framework that:

- preprocesses the video feed from Kuri in real-time;
- organizes the data and stores the prepossessed data in document-oriented database;
- performs successive refinement of the data to update data in the database;
- uses a subset of preprocessed data along with the query to be directed to original frames that need to be reprocessed for addressing the query for which data is not present in the database.

II. Related Work

Paul et al. [20] worked on extending the space of commands that a robot can understand. However, their system only incorporates factual information from past visual observations and linguistic interactions for around five minutes of data. In their system, increasing the length of the videos increases the context for inference, but it also increases the chances of failures due to errors in perception. We are working on developing a system that performs successive post-processing of the input sensor data as discussed in section-III to increase the certainty of the stored information and reduce the amount of reprocessing needed to answer the queries.

In recent years, computer vision has achieved significant success both in terms of accuracy and efficiency for object detection [1] and face identification [19]. Researchers have come up with various object detection algorithms such as Mask-RCNNs [9], RetinaNet [15], and further efforts have been made to make these convolution networks even faster - [8], [21], [4]. Likewise, learning deep video-representations (features) for activity recognition via convolution network has been receiving increasing attention [25], [6], [2], [13]. These deep networks have achieved high recognition performance in a variety of action datasets [14]. RoboMem will be using these state of the art techniques to pre-process data in real time for extracting necessary information from the video sensor data.

Chung* et al. [3] discusses question answering systems for autonomous mobile robots. Their system stores static and dynamic information of the indoor office environment in a world map and uses that to answer both information

acquisition and information retrieval queries, however, their system does not handle questions that span over a period of time and those involving object search. Our framework of RoboMem tries to incorporate both of these functionalities.

III. Technical Approach

In this section, we discuss the proposed system architecture of RoboMem that will enable robots to have long term memory and answer queries related to the healthcare of elderly and their surroundings. The architecture is also shown in Figure-2. In our model, RoboMem receives input - video feed and pose information from SLAM navigation which it preprocesses in real time to (i) store preprocessed data into database and (ii) store compressed video stream on external storage The preprocessing that we choose for RoboMem is explained in Section-IV-A. We envision preprocessing to include object detection on every frame and activity recognition model running on the frames in which a human is detected. Successive refinement of this preprocessed data needs to be performed to achieve higher accuracy and robustness [12], [5]. RoboMem will accomplish this by passing the preprocessed data through clean-up wizards. The post-processing that will be performed by the clean-up wizards include:

- Maintaining probabilities of the
  - Activities recognized and performed by the elderly
  - People and objects involved in the activity
  - Location of where the activity was performed
- Updating the probability distribution of various features such as the location of objects over different frames over time;

For updating probabilities, these clean-up wizards shown in Figure-2 will process combined data from multiple frames. This additional information gained will help increase the certainty of the data stored. To handle the case when sufficient data is not present in the database at the time when the query is asked, RoboMem will include a module for reprocessing the relevant video stream frames. A real-life example of this can be when elderly inquires from RoboMem about *"Hey Kuri, can you tell me the days when my grandson was wearing red T-shirt?"* and RoboMem's preprocessed data does not include the color of the T-shirts stored and will just have information of when the grandson was present. To handle this situation, RoboMem will fetch relevant frames that include the grandson and then reprocess frames to extract the color of the T-shirt.

A means of accessing long-term memory of RoboMem is query processing. The eventual goal is to enable RoboMem to understand natural language queries by all elderly, caregivers, and the doctors. This includes translation of the natural language query into an intermediate form - a formal query representation: $Query'$ (Figure-2) which can then be converted into internal DB commands using a method similar to [26]. These DB commands will then be used to fetch the attributes with the highest probability which will be used to create a response to the query in natural language. As mentioned previously, these probabilities are stored and being updated in the database. Technical challenges involving language grounding of the queries is discussed in the section-IV.

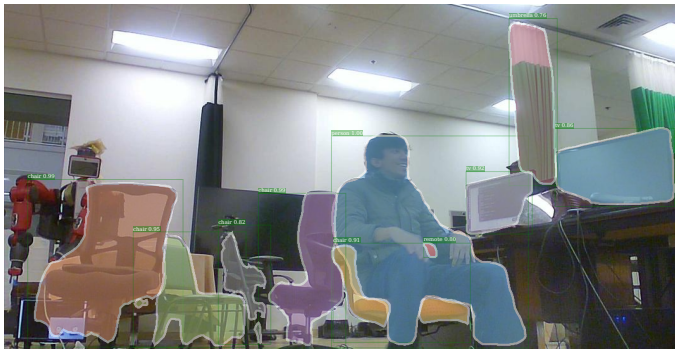

Fig. 3: Segmented objects in Kuri's image frame

RoboMem will also be exploring ways to store real-world knowledge since it will be integrating static information about the real world, elderly's life and their surrounding to answer some of the queries. For example, for RoboMem to understand queries like a doctor asking from RoboMem: "Hey Kuri, has my patient taken her medicines this week?", RoboMem needs the information of what medicines have been prescribed to the elderly to be able to ground queries and then respond to them accordingly. Suh et al. [23] proposes a multi-level ontology model for storing real-world knowledge. Different information about the environment (*e.g.,* texture of objects, activities associated with an object) is saved in different ontology layers and answers are obtained using goal based query reasoning while Topp et al. [24] represents the environment as a hierarchical graph modeled using a tour-guide like interaction between human(guide) and the robot. Chung* et al. [3] stores a static 2-D map of the real world, which is subdivided into regions and offices with information of the person to whom the office is assigned.

IV. Technical Challenges

In this section, we describe various validity and scalability issues that need to be addressed before RoboMem can robustly and efficiently answer queries by doctors, caregivers about the elderly's healthcare, and their surrounding. These challenges are as follows:

IV.A. Massive Sensor Data

RoboMem receives continuous raw video feed data from sensors as input. If preprocessing of this data is not performed in real-time, frames yet to be processed will continually accumulate and in the worst case they can gather to such an extent that even an overnight preprocessing of this video feed might not help to add relevant information in the database before a related query is asked leading to time-delays in answering them. Further, at the time when a query is asked, RoboMem cannot reprocess all of the sensor data when asked a query since it will hinder the process of answering the queries in a timely fashion. Therefore, RoboMem needs to be designed such that it is able to preprocess sensor data in real-time and store basic information in the database that can be refined in

the background and is enough to answer queries in real-time or, when/if required, is able to restrict the number of frames needed to be reprocessed for responding to query.

We tackle this challenge in the initial study and evaluate the feasibility of our proposed system by developing a prototype that:

- Extracts information from the sensor data that is fundamental to answering queries.
- Validates that preprocessing of the video can be done in real-time;

Our prototype is described and evaluated in section-VI. Right now preprocessing is done for every frame, but RoboMem will later have to fine-tune variables as to whether processing should be done for every frame to speed up pre-processing. Further, we need to explore if this level of object detection by Detectron will be appropriate for the queries. We also need to note that stacking object detection and activity recognition will also yield real-time performance, since state-of-art activity detectors can process more frames per second than Kuri's current frame rate which is 6 frames per second [27].

IV.B. Memory Representation

In our design, both the compressed video stream and data in the database needs to be stored. Although cloud storage can be used to solve the problem of storing compressed video stream, but one cannot merely assume infinite memory. Therefore, with regards to storage of video stream, we can work on saving summaries of really old data. In this regard, we can extend the model of [16],[10] to our context. Mastrogiovanni et al. [16] applies the migration of memory items from short to medium and long term memory. Their model is demonstrated to work on simple situations, developing memory for 4 blocks on the table, we intend to use their work as a foundation and apply on a complex scenario where the environment is dynamic and objects to be detected are not limited.

Secondly, with respect to the database design RoboMem cannot store every information of every object detected in every scene. The database needs to be designed such that it stores basic data and can return a small set of frames which can be processed to derive an answer for the query. The structure of the database that we chose for the initial prototype is described and justified in section-VI-C. Our construction of database collection allows us to answer queries related to spatiotemporal localization of objects in the elderly's surrounding and identifying people elderly has interacted with. In the future, we will explore, should data be stored for each frame or only for the keyframes?

IV.C. Post-processing Layer Design

The information extracted from pre-processing is not going to be precise enough to answer queries. Therefore, RoboMem needs to have a post-processing layer to increase the certainty of data in the database. Clean-up wizards need to maintain consistency while performing this post-processing, as new sensor data is received it needs to update the probabilistic location of the detected objects or humans , *e.g.,* the location of the elderly in two frames could be different. Clean-up wizard needs to identify that the person in both the frames is the same elderly and update the information accordingly. For future work, we will be exploring which design of the post-processing layer leads to most efficient look-up for queries that it has not seen before.

IV.D. Cost of Reprocessing Data

The ability of RoboMem to have long-term memory and respond to queries about the elderly's health relies heavily on appropriate pre-processing as reanalysis of prior image data in the general case is likely to be very expensive. Further, the possibility of queries that can be asked from RoboMem regarding health-care monitoring of elderly even with the constraints of categories mentioned in Section-I is vast and therefore there will be situations where enough information is not in the database to adequately answer the query leading to the expensive operation of reprocessing the compressed video feed. As discussed above, an architecture that guides to relevant sample frames for reprocessing as opposed to reprocessing the whole video feed of years or even hours of data will help ease this challenge. This can include identifying which objects are important to consider, adding different type of image processing in the pre-processing queue and setting a hierarchy of priorities between them accordingly. Along with this, as an extension to our prototype, we believe that as future work a learning agent needs to be part of RoboMem that will learn from the past queries to adjust the pre-processing accordingly.

IV.E. Natural Language Grounding of Query

As discussed in [26], a RNN can be used to for the translation of natural language query to MongoDB commands, but this will require manually curated training corpus of natural language sentence - MongoDB commands pair. Even after the translation, one might have to perform post-processing such as DB command correction(finding time-spans for periods like *"Past month"*), or handling of compositional DB commands like aggregations. Further, once RoboMem fetches response for a query related to the elderly, the response will comprise of multiple frames with different associated probabilities. In the future work, we will be exploring ways of grounding natural language queries to fetch responses from the database.

V. Future Work

Currently, our prototype processes information in marginal real time for 37 minutes of video as discussed in section-VI-D. However, as the amount of data increases, improvements need to be made in processing power and techniques. As a part of future work, we intend to address all the challenges mentioned in section-IV and build an end-to-end system. We hope to return a small set of relevant frames for visual processing in the next eight months. Next, we will work on grounding natural language query instead of working with intermediate queries for our system for another year or so.

Query Types	Query Examples	DB Interaction	DB Command
Spatio-temporal object localization	*Hey Kuri, where did you last see Ifrah?*	Yes	db.find($\{'label': 'ifrah'\}$).limit(1) .sort()$['location\_of\_object']$
Identifying the people elderly has interacted with	*Hey Kuri, Did Steve visited me yesterday?*	Yes	*db.find*$\{"label" : "Steve", 'timeStamp' : newDate(current\_date - 1, time : 00 : 00am), \$lt : newDate(currentdate)\}$
Was the activity performed?	*Hey Kuri, has my dad taken the Montelukast medicine over the past month?*	No	
How long was the activity performed	*Hey Kuri, how much has my patient slept over the last 3 months?*	No	
Where was the activity performed	*Hey Kuri, where did my patient exercise or walk the most over past week?*	No	

TABLE I: DB commands for queries handled by intial prototype

VI. Initial Feasibility Study

VI.A. Extraction of Data

For our prototype of RoboMem, we allow the Kuri Robot [22] from Mayfield Robotics to make observations and collect video in an indoor environment setting. We collect 37 minutes of video data with 13320 image frames.

Kuri has 2 RGB-D cameras that capture video feed with a frame rate of 6FPS. For our purposes we are using image stream from the left eye camera. The default size of the images is 1920*1080 pixels, but for speeding up the preprocessing, we resize the images to 1067*600. We also extract Kuri's pose estimate – x,y,z – in meters and roll, pitch, yaw of the camera in degrees. We are extracting this sensor data for identifying objects and estimating their location.

VI.B. Prototype's preprocessing sequence

Our prototype focuses on extracting the basic data needed to enable further post-processing or reprocessing to answer two types of queries that can be asked from RoboMem by caregivers, doctors or elderly as discussed in section-I – Spatiotemporal localization of objects in the surrounding of elderly and identifying the people elderly has interacted with. Our prototype performs real-time object detection by using Facebook's AI Research software system - Detectron [7]. Detectron includes implementation of multiple object detection algorithms, and for our implementation of RoboMem we use an end-to-end trained Mask R-CNN model with a ResNet-101-FPN backbone from a model zoo trained by Girshick et al. [7]. Once the objects are detected in frame f, the objects that are humans are labeled manually. Figure-3 shows the segmentation of objects that we achieve for a sample image frame captured in the 37 minutes of video.

The reason we do this kind of preprocessing is to store minimal information in DB for answering different query categories, as mentioned in section-I, regarding elderly healthcare. For each of those categories, we need to perform the fundamental operation of detecting objects that are of primary focus in the query. Take, for example, the lost object identification query by the elderly: *Hey Kuri, Do you remember where I last placed the Television's remote?* For this query, at the basic level, RoboMem needs to detect all the video frames that contain the remote that our prototype is able to detect. This detection is done as a preprocessing step. Another example query for identifying important people can be *"Hey Kuri, can you tell me if Steve visited me last summer?"* This will require RoboMem to detect frames captured within a time frame(last summer) that contain a person labeled as Steve in the database. Through object detection and the extraction of time and robot pose, our prototype can provide us information that is fundamental for answering these queries. RoboMem will later have to fine-tune variables as to whether pre-processing should be done for every frame to speed up pre-processing.

VI.C. Database Design

For queries related to spatiotemporal localization of objects and identifying people elderly has interacted with - the fields that our prototype integrates in the database collection are:

- Image frame number say f;
- Objects detected in the frame f;
- Object labels for objects detected e.g., name of the human;
- Robot's pose - x,y,z,roll,pitch,yaw - at frame f;
- Timestamp when Kuri captured frame f;
- An estimate of the location - x, y coordinates of the objects detected. We assume that the objects detected in the image frame are at the center of a circle within the radius of 2m. Our future work includes a more accurate localization of objects.

The reason behind choosing such a design structure for the initial database collection is that in the case of lost object identification, our prototype with such a design can return key information like the latest timestamp for the frame in which it saw the object. Likewise, for the identification of important people, it returns whether it has seen that specified person - the person named Steve - in the database. RoboMem is able to respond to such queries, without having to reprocess the whole video feed by having a structure that stores minimal information in the database.

We implement our prototype for RoboMem using python API for MongoDB - pyMongo. The choice to use MongoDB over the other databases was because of its flexibility to augment fields in the collections, which will be useful because of our multi-tier processing. Furthermore, in the future we intend to expand our framework to cloud databases, and MongoDB provides additional benefits of supporting distributed systems, allowing us to handle our application at scale [11].

VI.D. Quantitative Evaluation of prototype

The image frames are sent from Kuri to a standalone desktop for image processing. The time taken to transfer these images can be ignored because of the high speed network connection. Detectron has an inference time of 0.143 seconds per image, on average. The total time to organize preprocessed data for 13320 frames and inserting it in the mongoDB requires 59.382 seconds while the average time to insert a document with 6 fields mentioned in section VI-C in the database was 0.0041 seconds. Thus, with more sophisticated preprocessing, e.g., extracting depth of the objects, we want to be able to insert preprocessed data in the database with improved performance in real-time.

The size of the actual video and metadata (robot pose and timestamps) for the jpeg frames was 535.8MB while the size of 37 minutes of video feed encoded in mp4 was 79.5 MB, compared to the database size of only 3.5MB. These numbers not only show the amount of space we save after preprocessing the data but also that storing the feed as a video rather than individual images is much more scalable.

Table-I shows the DB commands that we use in our prototype to fetch data for the queries that we intend to address, as mentioned in Section-I.

VII. Conclusion

We design a query framework for robots to enable monitoring health-care of elderly and their surrounding environment by responding to queries over long, continuous feed of sensor data. We discuss the technical challenges in enabling robots to develop long-term memory and draft an architecture to overcome these challenges by performing i) real-time preprocessing of massive video sensor data (ii) successive refinement of data in database performed by clean-up wizards, and augmentation of this data into the database; (iii) reprocessing of data if enough information is not in the database; (iv) probabilistic language grounding for query processing; (v) learning from past queries to adjust the prepossessing queue and/or structure of MongoDB collections. We also conduct an initial study to show the feasibility of our proposed system. We believe that with these aforementioned components, RoboMem will be able to overcome the limitations involved in giving long term memory to robots and help provide caregivers and the elderly with information about the elderly and their surrounding.

References

[1] Lo-Bin Chang, Ya Jin, Wei Zhang, Eran Borenstein, and Stuart Geman. Context, computation, and optimal roc performance inhierarchical models. *International Journal of Computer Vision*, 93(2):117–140, Jun 2011. ISSN 1573-1405. doi: 10.1007/s11263-010-0391-1. URL https://doi.org/10.1007/s11263-010-0391-1.

[2] Guilhem Cheron, Ivan Laptev, and Cordelia Schmid. P-cnn: Pose-based cnn features for action recognition. pages 3218–3226, 12 2015. doi: 10.1109/ICCV.2015.368.

[3] Michael Jae-Yoon Chung*, Andrzej Pronobis*, Maya Cakmak, Dieter Fox, and Rajesh P. N. Rao. Autonomous question answering with mobile robots in human-populated environments. In *Proceedings of the 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Daejeon, Korea, October 2016. doi: 10.1109/IROS.2016.7759146. URL http://www.pronobis.pro/publications/chung2016iros.

[4] Jifeng Dai, Yi Li, Kaiming He, and Jian Sun. R-FCN: object detection via region-based fully convolutional networks. *CoRR*, abs/1605.06409, 2016. URL http://arxiv.org/abs/1605.06409.

[5] Justin Downs. Multi-frame convolutional neural networks for object detection in temporal data, 2017. URL https://calhoun.nps.edu/handle/10945/52976.

[6] Christoph Feichtenhofer, Axel Pinz, and Richard P Wildes. Spatiotemporal multiplier networks for video action recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4768–4777, 2017.

[7] Ross Girshick, Ilija Radosavovic, Georgia Gkioxari, Piotr Dollár, and Kaiming He. Detectron. https://github.com/facebookresearch/detectron, 2018.

[8] Ross B. Girshick. Fast R-CNN. *CoRR*, abs/1504.08083, 2015. URL http://arxiv.org/abs/1504.08083.

[9] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross B. Girshick. Mask R-CNN. *CoRR*, abs/1703.06870, 2017. URL http://arxiv.org/abs/1703.06870.

[10] W. C. Ho, K. Dautenhahn, M. Y. Lim, P. A. Vargas, R. Aylett, and S. Enz. An initial memory model for virtual and robot companions supporting migration and long-term interaction. In *RO-MAN 2009 - The 18th IEEE International Symposium on Robot and Human Interactive Communication*, pages 277–284, Sep. 2009. doi: 10.1109/ROMAN.2009.5326204.

[11] MongoDB Inc. Mongodb and mysql compared, January 2019. URL https://www.mongodb.com/compare/mongodb-mysql.

[12] MJ Jones, A Broad, and TY Lee. Recurrent multi-frame single shot detector for video object detection. 2018.

[13] Amlan Kar, Nishant Rai, Karan Sikka, and Gaurav Sharma. Adascan: Adaptive scan pooling in deep convolutional neural networks for human action recognition in videos. *CoRR*, abs/1611.08240, 2016. URL http://arxiv.org/abs/1611.08240.

[14] Yu Kong and Yun Fu. Human action recognition and prediction: A survey. *CoRR*, abs/1806.11230, 2018. URL http://arxiv.org/abs/1806.11230.

[15] Tsung-Yi Lin, Priya Goyal, Ross B. Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. *CoRR*, abs/1708.02002, 2017. URL http://arxiv.org/abs/1708.02002.

[16] Fulvio Mastrogiovanni, Ferdian Pratama, and Nak Chong. Long-term knowledge acquisition using contextual information in a memory-inspired robot architecture. *Journal of Experimental and Theoretical Artificial Intelligence*, 02 2016. doi: 10.1080/0952813X.2015.1134679.

[17] Robotics Mayfield. Meet kuri! the adorable home robot, 2018. URL https://www.heykuri.com/.

[18] C. Mucchiani, S. Sharma, M. Johnson, J. Sefcik, N. Vivio, J. Huang, P. Cacchione, M. Johnson, R. Rai, A. Canoso, T. Lau, and M. Yim. Evaluating older adults' interaction with a mobile assistive robot. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 840–847, Sep. 2017. doi: 10.1109/IROS.2017.8202246.

[19] Omkar M Parkhi, Andrea Vedaldi, Andrew Zisserman, et al. Deep face recognition. In *BMVC*, volume 1, page 6, 2015.

[20] Rohan Paul, Andrei Barbu, Sue Felshin, Boris Katz, and Nicholas Roy. Temporal grounding graphs for language understanding with accrued visual-linguistic context. *CoRR*, abs/1811.06966, 2018. URL http://arxiv.org/abs/1811.06966.

[21] Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. Faster R-CNN: towards real-time object detection with region proposal networks. *CoRR*, abs/1506.01497, 2015. URL http://arxiv.org/abs/1506.01497.

[22] Ian Richardson. Genesis of kuri, Mar 2018. URL https://futurism.media/genesis-of-kuri.

[23] Il Hong Suh, Gi Hyun Lim, Wonil Hwang, Hyowon Suh, Jung-Hwa Choi, and Young Tack Park. Ontology-based multi-layered robot knowledge framework (omrkf) for robot intelligence. pages 429 – 436, 10 2007. ISBN 978-1-4244-0912-9. doi: 10.1109/IROS.2007.4399082.

[24] E. A. Topp, H. Huettenrauch, H. I. Christensen, and K. S. Eklundh. Bringing together human and robotic environment representations - a pilot study. In *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4946–4952, Oct 2006. doi: 10.1109/IROS.2006.282456.

[25] Du Tran, Lubomir D. Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. C3D: generic features for video analysis. *CoRR*, abs/1412.0767, 2014. URL http://arxiv.org/abs/1412.0767.

[26] Prasetya Utama, Nathaniel Weir, Fuat Basik, Carsten Binnig, Ugur Çetintemel, Benjamin Hättasch, Amir Ilkhechi, Shekar Ramaswamy, and Arif Usta. An end-to-end neural natural language interface for databases. *CoRR*, abs/1804.00401, 2018. URL http://arxiv.org/abs/1804.00401.

[27] Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. Temporal segment networks for action recognition in videos. *IEEE transactions on pattern analysis and machine intelligence*, 2018.