Quantile QT-opt for Risk Aware Vision Based Robotic Grasping

Tilak Gurappagari
Graduate Student, Dept of EECS, Wichita State University
Wichita, KS, United States of America
Email: txgurappagari@shockers.wichita.edu

Abstract: *Reinforcement Learning(RL) based on Q learning algorithms are used in the state-of-the-art performance in arcade game environments. In this paper, a QT-Opt distributed Q-learning algorithm is presented and analyzed its conduct in a progression of simulated and real vision based robotic grasping.*

In this paper, a distributional upgrade of QTOpt subbed Quantile QT-Opt (Q2-Opt) presented distributed Q-learning algorithm that works on continuous action spaces, is directed by its exhibited pertinence to huge scope vision-based robotic investigations. Moreover, by being a sans actor speculation of Q-learning in continuous action spaces, QT-Opt empowers an immediate correlation with the past outcomes on the arcade conditions without the extra complexities and exacerbating impacts of an actor-critic-type engineering. present two adaptations of Q2-Opt, based on Quantile Regression DQN (QR-DQN) and Implicit Quantile Networks (IQN). The two strategies are assessed on a vision-based grasping undertaking in reproduction and the real world. We show that these distributional algorithms accomplish state-of-the-art grasping achievement rate in the two settings, while additionally being more example effective. The interactions between the robot and the environment are either put away in a database of episodes for later offline learning, or they are straightforwardly shipped off the replay cushion when online learning is performed. The examples from the Simulation Buffer are pulled by the Bellman Updater, which affixss the distributional targets. These marked changes are pushed to the train support and devoured by the TPU Training laborers to register the gradients. The boundary worker utilizes the gradients to refresh the weights, which are asynchronously pulled by the agents.

There are similar architectures for Q2R-Opt and Q2F-Opt. For Q2R-Opt, we modify the output layer of the standard QT-Opt architecture to be a vector of size N = 100, rather than a scalar. In simulation, the agent is prepared to grasp from a bin containing 8 to 12 randomly created procedural item. The scripted policy is bringing down the end effector at a random situation at the degree of the bin and endeavors to grasp. After 5, 000 steps, we switch to an -greedy policy which is equal to 0.2. We train the network from scratch using Adam with a learning rate 10^{-4} and batch size 4096 (256 per chip on a 4×4 TPU). Two iterations of CEM with 64 samples for each, the model is trained offline from a 72 TiB dataset of real-world experiences collected over five months, containing 559, 642 episodes of up to 20 time steps each. Out of these, 39% were generated by noise-free trained QT-Opt policies, 22% by an -greedy strategy using trained QT-Opt policies and 39% by a greedy strategy based on a scripted policy.

Project procedure and results: In this, we propose to get the hang of grasping by a humanoid robotic hand utilizing just the natural-language portrayals of an object - something like to requesting that a blindfolded individual grasp an object. Individuals disguise smoothness by rehashed communication with objects. Our methodology is to learn robotic grasps by imitating this human learning process. A vital thought of this investigation is to discretise the 10 degrees-of-freedom (DOF) joint design space into 6 unmistakable human grasp type subspaces, whereby just the grasp type and scale should be scholarly. This methodology decreases the problem dimensionality and renders it into a multi-class classification problem of choosing one possible human like grasp. The accomplishment of the methodology is assessed by scoring the predicted grasps against the human named grasps and furthermore by executing the grasps on an AR10 robotic hand with new arrangement of objects.

Forward kinematics alludes to the utilization of the kinematic equations of a robot to figure the situation of the end-effector from determined qualities for the joint parameters. The opposite cycle that figures the joint parameters that accomplish a predetermined situation of the end-effector is known as inverse kinematics.

Simulation Environment: The simulation is performed in Window OS. The hardware setting is CPU i7-7500U.The program used for the simulation is the pybullet library under the python environment.

Understanding Human Grasps: Most of the efforts in understanding grasps have been to breakdown the human grasping actions into discrete classes. A structured classification of grasps is discussed in based on object shapes and task requirements. Recently, a more comprehensive version of the grasp taxonomy has been developed by de-coupling them from the object shapes and the tasks being performed.

Learning Robotic Grasps: Robotic dexterity has been a difficult goal for a while and multiple approaches have been proposed to help robots master the grasping skill. Earlier methods involved analytical approaches to calculate object affordances and contact forces to determine grasp successes.

Natural Language Processing: Specific to robotics, natural language descriptions to understand object affordances, have been studied but mostly in the context of complementing machine vision and to recognize objects.

Grasp Taxonomy: Most studies attempting to understand and codify human grasps have come to conclusion that human grasp choice is a function of object affordances and the task requirements. For one specific object/task combination, there could be multiple grasp choices possible. However, human grasp choices do tend to cluster when studied over a large set of objects.

Process of Grasping : The process of grasping another is shown in figure 3. Grasping is usually preceded by several tasks that effect the final grasping action. The sequence of steps involved are:

A. The movement of the end-effector from a given position to within a reaching position from the object.

B. The estimation of grasp points and orientation of the end-effector to perform the grasp operation.

C. The grasping action, once the end effector is in the appropriate position.

RESULTS:

Grasping of object number 22

==========

handReading = [0.2281192330977454, 0.9775127968516315, 1.490715418090687, 0.22769563518322702, 0.976513990351622, 0.9674643162587586, 1.5699927786397827, 1.5671177911830372, 0.20067893789191849, 1.5696031100679266, 1.5698995421788833, 0.17001419659634018, 1.5699989178922256, 1.5698548089136473, 0.16999335590573206, 1.5675524549546094, 1.5703974740118485, 0.16999645995741208, 1.5696524974879926, 1.569852315324604, 0.1700114048201699, 1.5698961680343697, 1.5690251453094404, 0.17007276550393854, 1.5710353778988084, 1.4999866738951975, 1.4998503192803874]

==========

orientation = [1.0247223377227783, 0.9426326314595084, 0.9261510372161865]

==========

palmPosition = [0.85, -0.05, 0.25]

==========

==========

==========

==========

==========

PalmCOntact: N/A (Object did not come into contact with palm)

==========

thumbContact:

[62, (0.8447918118202521, -0.034289953999756484, 0.2621253672622273), -0.00019771481018187158, 63.1568169210738, -25.235641491746748, (-0.15507068273148483, -0.9879033775411395, 0.0), -18.983105104290456, (0.14172474894955361, -0.022246460614656984, 0.989656097099041)]

[62, (0.8442222556932008, -0.03130740058332152, 0.27574989113842185), -1.1275060518276682e-05, 99.65884250688842, 49.535489216452355, (0.0, 0.9179705778535664, -0.39664848190203583), 5.404306665977596, (0.8396573221460865, -0.2154263530386249 8, -0.498565008582572 8)]

==========

indexContact:

[53, (0.8493859179960342, -0.03742511460960314, 0.2988233978541963), 0.0005977626779464977, 0.0, 0.0, (0.2333958882041118, 0.9723817971195337, 0.0), -0.0, (-0.4507208481516582, 0.10818424717337156, 0.8860851458550589)]
[54, (0.8503069078359649, -0.017769484217157013, 0.2966143370127234), -1.383879059471097e-05, 7.604198128732139, -2.063293757656842, (-0.8919621761370116, 0.4521100268086593, 0.0), -3.1935522799646874, (0.2146820649127865, 0.42354354126774835, 0.8800695879617461)]
[48, (0.8536226798922082, -0.03942364480520781, 0.2910866552893122), -0.00093702042792946, 0.0, -0.0, (0.0, -0.9803927674588911, 0.19705334687413134), -0.0, (0.9994747009625098, 0.006386237635259673, 0.03177322937266125)]
[48, (0.8534767750352829, -0.04115080123006186, 0.29184619799771716), -0.001586577084480609, 51.280226668637674, -18.62477422759746, (0.17195938607175912, 0.9851040399581275, 0.0), 1.254377706175291, (-0.6472360900195748, 0.11298128539598999, 0.7538704616354476)]
[48, (0.8525895905317973, -0.035301053410859935, 0.2911640752357072), 0.0009140148547008552, 0.0, 0.0, (0.26080270321278226, 0.9653921224025528, 0.0), -0.0, (-0.3340948715992392, 0.0902564290930437, 0.9382080759503514)]
[48, (0.8502120143515398, -0.041471488858012974, 0.2929837720986795), 0.001015765440906578, 0.0, 0.0, (0.31309688873360186, 0.9497211897527289, 0.0), -0.0, (-0.06023112444683519, 0.019856540922436862, 0.9979869385069454)]
[53, (0.8489874294853668, -0.03599730099402522, 0.3001134860207897), 0.0008375011515451194, 0.0, 0.0, (0.26219849352011004, 0.9650139636273585, 0.0), -0.0, (-0.33413871817510554, 0.09078694385202106, 0.938141272859427 9)]
[53, (0.8495193897820161, -0.03589063058073597, 0.29879460148893205), 0.0008469676294082539, 0.0, 0.0, (0.26030456729836965, 0.9655265569851555, 0.0), -0.0, (-0.33420400605238065, 0.09010091804878667, 0.9381841540472245)]
[48, (0.8519389936065092, -0.042753242403069824, 0.2915216205091216), -0.0004259391274455351 5, 0.0, 0.0, (0.22778135879851724, 0.9737123048333636, 0.0), -0.0, (-0.47359495326558465, 0.11078847565084077, 0.873746950726772)]
==========
midContact:
[39, (0.8630413904323068, -0.042556896316702945, 0.27548482073968833), 0.0009874658827866722, 0.0, -0.0, (0.4205641023583911, 0.9072628262016916, 0.0), -0.0, (0.45583322454442066, -0.21130270674513144, 0.8646197068790513)]
[44, (0.8613276009709576, -0.04224508705956839, 0.2830749077696052), -0.0007336245539068825, 0.0, -0.0, (0.33359582953812456, 0.9427161940450428, 0.0), -0.0, (0.04877599937250076, -0.017260199914888686, 0.9986605966914446)]
[44, (0.863448810423234, -0.039959187469300085, 0.28190460301556874), -0.0018956871925798371, 107.02071511874435, 26.732221216095567, (0.0, -0.9995113145924115, -0.031259110699919046), 14.461747399922045, (0.8427067803361287, 0.01682905526062951, -0.5381097149044518)]
==========
ringContact:
[31, (0.8550469495120823, -0.02989781810924883, 0.2409653498539029), 1.6758293562592318e-06, 4.789109120089891, -0.9131355751421099, (-0.9707572358344791, 0.24006330222464575, 0.0), -2.2136112943427833, (-0.014590780761734829, -0.059001546132503635, 0.9981512544052316)]

==========
pinkyContact: N/A (Object did not come into contact with pinky finger)

Grasping of object number 23
handReading = [1.5703939817235741, 1.5170506695556734, 0.3942118426107321, 1.5701513099328068, 1.5676584721310547, 0.3197565791288182, 1.5700353933102624, 1.5643535387075613, 0.21099075957800825, 1.5688908930569812, 1.5675893925658642, 0.1708699225516982, 1.5702504275317464, 1.5524657778091588, 0.16409665095620732, 1.5696558259746816, 1.5695026140467518, 0.17080954665981496, 1.569873887634572, 1.5700071405521063, 0.17128414429860384, 1.5685309004163706, 1.5699971904184533, 0.17092139121952332, 1.5647712502938942, 1.4510790404899163, 1.4870055843403438]
==========
orientation = [-0.06656789779663086, -0.2147962292048593, -0.0989999771118164]
==========
palmPosition = [0.85, -0.05, 0.25]
==========
==========
==========
==========
PalmCOntact:
[19, (0.8632677481658734, -0.05485157847303272, 0.2023078700293872), -0.004782592550450239, 185.61957011222347, 36.92239696124871, (0.8953460792000568, -0.4453710795068374, 0.0), 33.849014017244585, (-0.10643287642775102, -0.21396597801790598, 0.9710255419226397)]
[19, (0.8245667371813787, -0.04831281194227052, 0.23438978918390385), -0.007782005253181359, 223.65015183349638, 5.7623203606767675, (0.9273094718585604, -0.3742955294969443, 0.0), 16.434785986671542, (-0.07149905906758976, -0.17713744749091895, 0.9815855588021104)]
[19, (0.8206586437329914, -0.03884154635711719, 0.23554137179461765), -0.000376808031956806, 178.18846591994586, 62.35518707672107, (0.927309471858392, -0.3742955294973615, 0.0), 63.636569607050966, (-0.0714990590677872, -0.17713744749117846, 0.9815855588020492)]
[19, (0.8050304748546457, -0.026198559754175926, 0.20740865121238797), 0.0003243185142585861, 0.0, 0.0, (0.9542626562965294, -0.2989695348992801, 0.0), -0.0, (-0.04354045843248669 6, -0.13897413839892028, 0.989338373528373 1)]
[19, (0.8274751109865999, -0.043934087879868106, 0.24286470721566433), -0.0007451012364175414, 156.5415665690368, 11.323738760477163, (0.900173639704451, -0.43553119105437405, 0.0), 75.55111607009691, (-0.10348657413229939, -0.21389027470500122, 0.9713606330096867)]

[19, (0.811218739866992, -0.029700794215378426, 0.21804659922574257), 0.001152192624022885, 0.0, 0.0, (0.9370427829874114, -0.349214579522229, 0.0), 0.0, (-0.0634431374054842, -0.17023611684260445, 0.9833588525246011)]
[19, (0.7895355649324169, -0.025409989211733448, 0.22846779026774977), -0.0004467769398642508 5, 0.0, 0.0, (0.942098870023171, -0.33533523390938874, 0.0), 0.0, (-0.057598706311850024, -0.16181919060092873, 0.9851380302193504)]
[19, (0.7990495497769644, -0.02919977923686655, 0.23064873057691648), -0.00027541751783139 7, 0.0, 0.0, (0.9302509952572523, -0.36692381473936997, 0.0), 0.0, (-0.08189152132058407, -0.2076171296368841, 0.9747763365086073)]
[19, (0.8056334148118809, -0.03181856414728422, 0.23213552837672655), -0.0001570558044261217 4, 0.0, 0.0, (0.9351493107725828, -0.35425381658009486, 0.0), 0.0, (-0.06805794204353668, -0.1796574507200096, 0.9813721602560295)]
==========
==========
thumbContact:
[61, (0.8128275299720596, -0.009465525110108734, 0.2042680093066103), 0.0003834505941488649 5, 0.0, 0.0, (-0.9544086148071694, 0.29850325958330864, 0.0), -0.0, (-0.03775013646659919, -0.12069903526066281, 0.9919711034520605)]
[59, (0.8033724570421805, -0.01540911392863607, 0.19367286246843413), -0.0007537550015561049, 0.0, -0.0, (0.0, 0.9994452914289865, -0.033303294738331005), -0.0, (0.9890727469722117, 0.0049098478516222885, 0.14734651197404447)]
[62, (0.820654233080681, -0.02483160161066843, 0.23596153036726145), -6.864453636139065e-05, 0.0, -0.0, (-0.4431052794791322, -0.8964695819143671, 0.0), -0.0, (-0.026169401797450996, 0.012934962135021873, 0.9995738337732382)]
[59, (0.8037395306481684, -0.01458592407456076, 0.19345038204297565), -0.0006930707450511997, 261.39803642187763, -128.15062210576681, (0.0, 0.9882591828700034, -0.15278673853811192), -25.683679938985872, (0.9889791946755437, -0.022620773225353417, -0.1463162769185626)]
[62, (0.8194112518231309, -0.02423904733102669, 0.23580251003709307), -0.0035455212663477 08, 317.0903610118821, -17.423337425339568, (-0.9702794731959454, 0.24198707381717455, 0.0), 6.547371240975124, (-0.11375408991933025, -0.4561122075644457, 0.8826220375320986)]
[64, (0.8213316121031936, -0.012388809386799144, 0.20564481213854655), -0.0012451243882814834, 148.70335253554146, 30.17669453326905, (-0.9515754567170006, 0.30741527316291883, 0.0), -67.95247509011831, (-0.049588802471375426, -0.1534975373027109, 0.9869039754259064)]
[59, (0.8029518112879398, -0.016118927094782524, 0.19397851832732663), -0.0003592377014054023 4, 50.756967470783714, -25.378429707198112, (0.0, 0.9998227894579423, -0.018825240517434654), 0.052367015890516525, (0.9059528314501042, 0.007970204602983864, 0.4233036062049623 7)]
[62, (0.820314998224423, -0.01710073620058758, 0.2261995842889902), -0.0047786756068848 51, 284.4335852704055, -4.614903342907149, (-0.9980445829608717, -0.06250608308364632, 0.0), 142.14189661946926, (0.0312252149740806, -0.49857798024178906, 0.8662822770713023)]
[64, (0.8203865525881526, -0.01535972986100258, 0.2192192694413861), -0.0002582672512122508, 0.0, -0.0, (-0.8684412552310918, 0.49579207961871075, 0.0), 0.0, (-0.17965944176746698, -0.3146957717490089, 0.9320349007559202)]
[64, (0.8198386987644737, -0.015273311817329816, 0.22083442214674678), -0.0003866143323588968, 0.0, -0.0, (-0.9145243205042279, 0.4045309224352082, 0.0), 0.0, (-0.06407629020627119, -0.1448574695070621, 0.9873755833326103)]

[64, (0.8206810327852039, -0.015886804626149297, 0.22238513455663955), -0.00040205419323832497, 0.0, -0.0, (-0.921669338597067, 0.38797632697117646, 0.0), 0.0, (-0.06000281855153612, -0.1425415785547727, 0.987968400379782)]
==========
indexContact: N/A (Object did not come into contact with index finger)
==========
midContact:
[42, (0.8297794870433096, -0.015357863987377006, 0.2070236555627107), 0.00014105624961379345, 104.38065431176108, 22.361702187533805, (0.9270613502573185, -0.3749096595969188 5, 0.0), 47.15701987948229, (-0.07045652037774149, -0.17422201654139002, 0.9821825531379147)]
[42, (0.831612144594876, -0.01603572133629881, 0.20741755895145336), 1.1902354187165149e-05, 0.0, -0.0, (0.9511620729768301, -0.30869193531807637, 0.0), -0.0, (-0.06169137509975825, -0.19008755821306086, 0.9798269716898476)]
==========
ringContact:
[33, (0.8500768732095509, -0.023891014457053376, 0.21857888743065063), 0.00015318993740016354, 0.0, -0.0, (0.9719962262250588, -0.2349964599823681, 0.0), -0.0, (0.038935628164193596, 0.1610461861601568, 0.9861785552235092)]
[33, (0.8385844145489666, -0.019148983817683707, 0.20857325292455406), 8.321230496168056e-05, 0.0, -0.0, (0.12896672665996847, -0.9916489214508394, 0.0), -0.0, (-0.10128783557022257, -0.013172767419393909, 0.9947699495682477)]
[33, (0.8508141114230653, -0.02300065022574863, 0.21071704513181758), -0.0019810486016248047, 0.0, -0.0, (0.9009237963840218, -0.4339773186573258, 0.0), -0.0, (0.05492913153375237, 0.11403121680776368, 0.9919574951086808)]
[33, (0.8516740521863924, -0.023298191029872448, 0.21089216619705412), -0.0020672317104966464, 67.80913486770571, -33.89638312041719, (0.9350107039424985, -0.3546194911633501, 0.0), -0.7449189422830734, (0.05084890415258886, 0.13407122521789472, 0.9896662546106407)]
==========
pinkyContact:
[24, (0.8688161564490555, -0.029296969491955083, 0.2142270350778574), -0.0024953276337470963, 0.07588638083038188, 0.026774241076691175, (0.8883582023812425, -0.45915106910685455, 0.0), 0.026885418234624144, (0.11820165449096971, 0.22869468540353002, 0.9662955602421953)]
[24, (0.8690354110185021, -0.02932440120316588, 0.2150003908721712), -0.002369968693480454, 209.45720467026646, 16.57565535567435, (0.8883582023812414, -0.4591510691068566, 0.0), 38.668144472258625, (0.11820165449097024, 0.22869468540352975, 0.9662955602421952)]

Conclusion: This paper has introduced a way to deal with parse object descriptions in natural language and decide the proper grasping technique utilizing the parsed object attributes. The framework of grasping methodology was created utilizing multiple machine learning models and performs sensibly well. Future examinations can take a gander at changing over remarkable grasps labels to multi-label grasp probability distribution to all the more likely speak to the human grasping conduct. The model can be scaled to incorporate distinctive

assignment conditions and a more extensive choice of objects. The neural network model could be effortlessly reached out to a deeper multi-layer network with extra highlights including object's position and direction.

Appendix

Source code:

For object 22:

```
self.objectId = p.loadURDF("random_urdfs/022/022.urdf", xpos, ypos, -0.1, orn[0], orn[1], orn[2], orn[3]) # line 98
```

For object 23:

```
self.objectId = p.loadURDF("random_urdfs/023/023.urdf", xpos, ypos, -0.1, orn[0], orn[1], orn[2], orn[3]) # line 98
```

main.py

```
#add parent dir to find package. Only needed for source code build, pip install doesn't need it.
import os, inspect
currentdir = os.path.dirname(os.path.abspath(inspect.getfile(inspect.currentframe())))
parentdir = os.path.dirname(os.path.dirname(currentdir))
os.sys.path.insert(0, parentdir)
import pybullet as p
import math

#from pybullet_envs.bullet.kukaGymEnv import KukaGymEnv
from sawyerEnv import sawyerEnv
import time

def main():

  # replace the values of following 3 variables to recall the last configurations
  # the values are the 3 output of the last run of this program
  handInitial = [0.22152291659747883, 0.9817759152539315, 0.16999407449044338, 0.2215143810144337, 0.9823135574132803, 0.1699989274565554, 0.5739483012567052, 0.5902239466515086, 0.1699767141257003, 0.5739640367999089, 0.5899904373506896, 0.17000733012736757, 0.9361331354216027, 0.26548528122516185, 0.17000335607201394, 0.93630346854007, 0.2656232807802453, 0.17001488487220198, 0.906821961784849, 0.25019764084658785, 0.17007600796776692, 0.906618391328536, 0.25115462875714, 0.16999999912072705, 1.5698416936818693, 0.3400976317456705, 0.3400749247584485]
  orientation =  [0, 0.5*math.pi, 0]
  palmPosition = [0.85, -0.05, 0.1]


  environment = sawyerEnv(renders=True, isDiscrete=False, maxSteps=10000000, palmPosition = palmPosition, orientation = orientation)
```

```
  readings = [0] * 35
  motorsIds = []

  dv = 0.01
  motorsIds.append(environment._p.addUserDebugParameter("posX", -dv, dv, 0))
  motorsIds.append(environment._p.addUserDebugParameter("posY", -dv, dv, 0))
  motorsIds.append(environment._p.addUserDebugParameter("posZ", -dv, dv, 0))

  # orientation of the palm
  motorsIds.append(environment._p.addUserDebugParameter("orienX", -math.pi, math.pi, 0))
  motorsIds.append(environment._p.addUserDebugParameter("orienY", -math.pi, math.pi, 0))
  motorsIds.append(environment._p.addUserDebugParameter("orienZ", -math.pi, math.pi, 0))

  #low [0.17 - 1.57], mid [0.34, 1.5]
  motorsIds.append(environment._p.addUserDebugParameter("thumbLow", 0.85, 1.57,
handInitial[24]))
  motorsIds.append(environment._p.addUserDebugParameter("thumbMid", 0.34, 1.5,
handInitial[25]))
  #[0.17 - 1.57]
  motorsIds.append(environment._p.addUserDebugParameter("indexLow", 0.17, 1.57,
handInitial[18]))
  motorsIds.append(environment._p.addUserDebugParameter("indexMid", 0.17, 1.57,
handInitial[19]))
  motorsIds.append(environment._p.addUserDebugParameter("middleLow", 0.17, 1.57,
handInitial[12]))
  motorsIds.append(environment._p.addUserDebugParameter("middleMid", 0.17, 1.57,
handInitial[13]))
  motorsIds.append(environment._p.addUserDebugParameter("ringLow", 0.17, 1.57,
handInitial[6]))
  motorsIds.append(environment._p.addUserDebugParameter("ringMid", 0.17, 1.57,
handInitial[7]))
  motorsIds.append(environment._p.addUserDebugParameter("pinkyLow", 0.17, 1.57,
handInitial[0]))
  motorsIds.append(environment._p.addUserDebugParameter("pinkyMid", 0.17, 1.57,
handInitial[1]))
  done = False
  action = []
  while (not done):

    action = []
    for motorId in motorsIds:
      action.append(environment._p.readUserDebugParameter(motorId))

    #print (action)
    #break
    #state, reward, done, info = environment.step2(action)
    state, reward, info = environment.step2(action)
    #environment.step2(action)
    #done = True
    #obs = environment.getExtendedObservation()
```

```
    handReading = environment.handReading()
    orientation1 = environment.o()
    palmPosition1 = environment.p()
    qKey = ord('q')
    keys = p.getKeyboardEvents()
    if qKey in keys and keys[qKey]&p.KEY_WAS_TRIGGERED:
      break;


print("============================================================
===============")
  print("handReading = ", handReading)

print("============================================================
===============")
  print("orientation = ", orientation1)

print("============================================================
===============")
  print("palmPosition = ", palmPosition1)

print("============================================================
===============")



if __name__ == "__main__":
  main()
```

References:

1. https://arxiv.org/abs/1910.02787
2. https://docs.google.com/document/d/10sXEhzFRSnvFcl3XxNGhnD4N2SedqwdAvK3dsihxVUA/edit#heading=h.2ye70wns7io3
3. https://ieeexplore.ieee.org/abstract/document/7243327?casa_token=cSsj6S7LefUAAAAA:Fp0HE0TpWuWNVEE3QnCTsUebTv1XHy3_RAHggA_xKtJHpnasr5EXXkLxz3ifRkT3I3mqFQ8