

# DOPE-Plus: Enhancements in Feature Extraction and Data Generation for 6D Pose Estimation

Jeffrey Chen

Rackham Graduate School (of UMICH)  
Department of Robotics  
Ann Arbor, United States  
jeffzc@umich.edu

Yuqiao Luo

Rackham Graduate School (of UMICH)  
Department of ECE  
Ann Arbor, United States  
joeluo@umich.edu

Longzhen Yuan

Rackham Graduate School (of UMICH)  
Department of ECE  
Ann Arbor, United States  
longzhen@umich.edu

**Abstract**—This study explored enhancements to the DOPE framework by improving both its network architecture and synthetic data generation pipeline for 6D object pose estimation. We proposed replacing the original VGG-19-based feature extractor with a Vision Transformer (ViT), aiming to leverage its superior representation capabilities. In parallel, we developed a refined data generation pipeline, resulting in an augmented HOPE dataset [1] and a new fully synthetic dataset of a customized object, Block. These datasets were used to train and evaluate our modified DOPE model on two target objects: Cookies and Block. Experimental results demonstrate that incorporating ViT improves pose estimation performance over the original VGG-19 backbone, suggesting the potential for further advancements through the integration of more powerful feature extractors. This project’s public repository is available at: <https://github.com/jypipi/DOPE-Plus>.

## I. INTRODUCTION

As robotics continues to advance, researchers are increasingly exploring ways to equip robots with the capabilities needed to perform everyday tasks. Many of these tasks require fundamental operations such as object fetching, which depend on accurate pose estimation of target objects. This study investigated the DOPE (Deep Object Pose Estimation) proposed by J. Tremblay et al. in 2018 [2], and further extended the feature extraction and data generation pipelines. The original DOPE framework employed VGG-19 as the feature extractor. In our work, we replaced it with a Vision Transformer (ViT) [3], [4], motivated by its superior feature extraction capabilities, particularly in capturing relationships between multiple objects. Meanwhile, we enhanced the data synthesis pipeline proposed by [2] to augment and generate two new datasets for network training. Our goal is to improve the accuracy of 6D object pose estimation and to validate the effectiveness of our enhancements for object perception in real-world scenarios.

### A. Original DOPE

DOPE (Deep Object Pose Estimation) is a one-shot, instance-based, deep neural network-based system designed to estimate the 3D poses of known objects in cluttered scenes from a single RGB image, in near real time and without the need for post-alignment. The system employs a straightforward deep network architecture trained entirely on synthetic data. It predicts the 2D image coordinates of the projected

3D bounding box corners, which are then used with the perspective-n-point (PnP) algorithm to recover the 3D pose.

1) *Network architecture*: The DOPE network is a convolutional deep neural network that detects objects’ 3D keypoints using multi-stage architecture. Firstly, image features are extracted by the first ten layers of the VGG-19 convolutional neural network (with pre-trained parameters). Then two  $3 \times 3$  convolutional are applied to the features to reduce the feature dimensions from 512 to 128. Second, these 128-dimensional features are fed into the first stage, which consists of three  $3 \times 3 \times 128$  convolutional layers and one  $1 \times 1 \times 512$  layer, followed by a  $1 \times 1 \times 9$  to produce belief maps and  $1 \times 1 \times 16$  to produce vector fields. There are 9 belief maps, 8 of them are for the projected vertices of the 3D objects and one for its centroid. Vector fields indicate that the direction from vertices to their corresponding centroids, to construct the bounding boxes of objects after detection. There are overall 6 identical stages as the first stage, except for the follow-up stages accept image features, belief maps and vector fields as their input, therefore they have five  $7 \times 7 \times 128$  and one  $1 \times 1 \times 128$  layers to align data before converting to belief maps and vector fields. The network could leverages increasing larger receptive fields as data go through the neural network. This enables the network reduce the ambiguities in early stages and thus produce context relationships in later stages.

2) *Data Generation*: As more data is required to train a deep network with high performance, it can be difficult to gather enough data for training. In addition, unlike 2D labeling, making 3D pose labels manually is much more difficult. DOPE proposed a method to generate data, which allows scientists to gather enough number of data rapidly, and greatly alleviate the workload of labeling manually.

The overall data synthesis strategy is to generate two kinds of dataset: "domain randomized (DR)" and "photorealistic (photo)". The domain randomized data are generated by putting the target object into a virtual environment, which is composed of different distractor objects and a random background. The objects shown in DR images do not necessarily obey physical principles. Photorealistic data are generated by putting target objects into 3D backgrounds with physical constraints. In other words, they are impacted by the effects of gravity and collision.

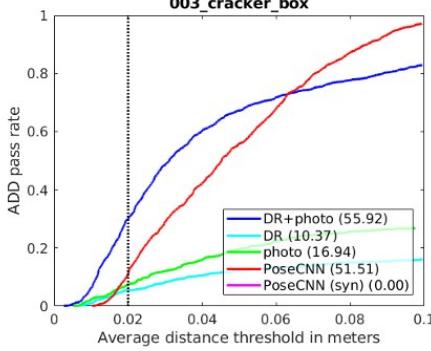


Fig. 1: Performance Comparison between DOPE and poseCNN on Cracker Box

Training on these synthesized data demonstrated a good experiment result. Figure 1 (reproduced from [2]) compares its performance with traditional poseCNN. It also shows that using a combination of DR and photo images gives a much better result than using them alone.

## II. RELATED WORK

Many efforts has been made to recognize object pose accurately. One of the most impactful work is poseCNN [5]. It can estimate the 6D pose from a single RGB image, and presents a decent result. Wang Y. et al. proposed DenseFusion [6], which further introduced depth information to enhance the model performance. In DOPE, the 3D model of the target object must be known beforehand to enable pose estimation, which helps improve precision. This approach is particularly suitable for applications such as fetching known objects in controlled environments. In general, various sensing modalities have been employed across different methods to maximize the robustness and accuracy of 6D object pose estimation.

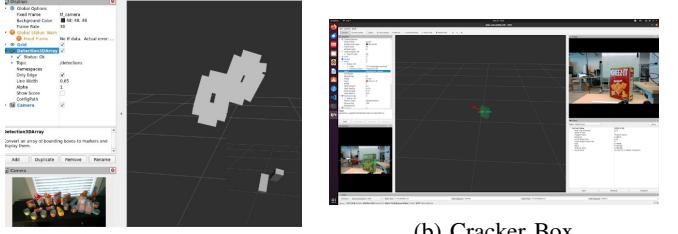
## III. REPRODUCTION & ALGORITHMIC EXTENSION

### A. Original Paper Reproduction

We reviewed the original DOPE paper and codebase to thoroughly understand the proposed network architecture, dataset compositions, and training techniques. To reproduce the original work, we configured our local computing environment with the necessary dependencies for the original framework. We successfully ran DOPE with the Robotics Operation System (ROS), establishing real-time data streaming and model inference in ROS Noetic. As shown in Figure 2, we deployed the pose estimation pipeline using pre-trained models and visualized inference results for two objects, Tomato Ketchup and Cracker Box, with both open-sourced datasets and Intel RealSense RGB-D camera streams, which were consistent with those reported in [2].

### B. Network Architecture

One of the key algorithmic extensions we introduced involves replacing the original VGG-19 feature extractor with a Vision Transformer (ViT). This modification was driven



(a) Tomato Ketchup  
(b) Cracker Box

Fig. 2: Pose Estimation Reproduction with RViz Visualization

```

1  # === 1. ViT backbone using timm ===
2  self.vit = timm.create_model(
3      'vit_base_patch16_224',
4      pretrained=True
5      , features_only=True
6  )
7
8  vit_out_channels = self.vit.feature_info
9      [-1]['num_chs']
10
11 self.feature_proj = nn.Sequential(
12     nn.Conv2d(vit_out_channels, 256,
13             kernel_size=1),
14     nn.ReLU(inplace=True),
15     nn.Conv2d(256, 128, kernel_size=1),
16     nn.ReLU(inplace=True),
17 )
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
279
280
281
282
283
284
285
286
287
288
289
289
290
291
292
293
294
295
296
297
298
299
299
300
301
302
303
304
305
306
307
308
309
309
310
311
312
313
314
315
316
317
318
319
319
320
321
322
323
324
325
326
327
327
328
329
329
330
331
332
333
334
335
336
337
337
338
339
339
340
341
342
343
344
345
345
346
347
347
348
349
349
350
351
352
353
353
354
355
355
356
357
357
358
359
359
360
361
361
362
363
363
364
365
365
366
367
367
368
369
369
370
371
371
372
373
373
374
375
375
376
377
377
378
379
379
380
381
381
382
383
383
384
385
385
386
387
387
388
389
389
390
391
391
392
393
393
394
395
395
396
397
397
398
399
399
400
401
401
402
403
403
404
405
405
406
407
407
408
409
409
410
411
411
412
413
413
414
415
415
416
417
417
418
419
419
420
421
421
422
423
423
424
425
425
426
427
427
428
429
429
430
431
431
432
433
433
434
435
435
436
437
437
438
439
439
440
441
441
442
443
443
444
445
445
446
447
447
448
449
449
450
451
451
452
453
453
454
455
455
456
457
457
458
459
459
460
461
461
462
463
463
464
465
465
466
467
467
468
469
469
470
471
471
472
473
473
474
475
475
476
477
477
478
479
479
480
481
481
482
483
483
484
485
485
486
487
487
488
489
489
490
491
491
492
493
493
494
495
495
496
497
497
498
499
499
500
501
501
502
503
503
504
505
505
506
507
507
508
509
509
510
511
511
512
513
513
514
515
515
516
517
517
518
519
519
520
521
521
522
523
523
524
525
525
526
527
527
528
529
529
530
531
531
532
533
533
534
535
535
536
537
537
538
539
539
540
541
541
542
543
543
544
545
545
546
547
547
548
549
549
550
551
551
552
553
553
554
555
555
556
557
557
558
559
559
560
561
561
562
563
563
564
565
565
566
567
567
568
569
569
570
571
571
572
573
573
574
575
575
576
577
577
578
579
579
580
581
581
582
583
583
584
585
585
586
587
587
588
589
589
590
591
591
592
593
593
594
595
595
596
597
597
598
599
599
600
601
601
602
603
603
604
605
605
606
607
607
608
609
609
610
611
611
612
613
613
614
615
615
616
617
617
618
619
619
620
621
621
622
623
623
624
625
625
626
627
627
628
629
629
630
631
631
632
633
633
634
635
635
636
637
637
638
639
639
640
641
641
642
643
643
644
645
645
646
647
647
648
649
649
650
651
651
652
653
653
654
655
655
656
657
657
658
659
659
660
661
661
662
663
663
664
665
665
666
667
667
668
669
669
670
671
671
672
673
673
674
675
675
676
677
677
678
679
679
680
681
681
682
683
683
684
685
685
686
687
687
688
689
689
690
691
691
692
693
693
694
695
695
696
697
697
698
699
699
700
701
701
702
703
703
704
705
705
706
707
707
708
709
709
710
711
711
712
713
713
714
715
715
716
717
717
718
719
719
720
721
721
722
723
723
724
725
725
726
727
727
728
729
729
730
731
731
732
733
733
734
735
735
736
737
737
738
739
739
740
741
741
742
743
743
744
745
745
746
747
747
748
749
749
750
751
751
752
753
753
754
755
755
756
757
757
758
759
759
760
761
761
762
763
763
764
765
765
766
767
767
768
769
769
770
771
771
772
773
773
774
775
775
776
777
777
778
779
779
780
781
781
782
783
783
784
785
785
786
787
787
788
789
789
790
791
791
792
793
793
794
795
795
796
797
797
798
799
799
800
801
801
802
803
803
804
805
805
806
807
807
808
809
809
810
811
811
812
813
813
814
815
815
816
817
817
818
819
819
820
821
821
822
823
823
824
825
825
826
827
827
828
829
829
830
831
831
832
833
833
834
835
835
836
837
837
838
839
839
840
841
841
842
843
843
844
845
845
846
847
847
848
849
849
850
851
851
852
853
853
854
855
855
856
857
857
858
859
859
860
861
861
862
863
863
864
865
865
866
867
867
868
869
869
870
871
871
872
873
873
874
875
875
876
877
877
878
879
879
880
881
881
882
883
883
884
885
885
886
887
887
888
889
889
890
891
891
892
893
893
894
895
895
896
897
897
898
899
899
900
901
901
902
903
903
904
905
905
906
907
907
908
909
909
910
911
911
912
913
913
914
915
915
916
917
917
918
919
919
920
921
921
922
923
923
924
925
925
926
927
927
928
929
929
930
931
931
932
933
933
934
935
935
936
937
937
938
939
939
940
941
941
942
943
943
944
945
945
946
947
947
948
949
949
950
951
951
952
953
953
954
955
955
956
957
957
958
959
959
960
961
961
962
963
963
964
965
965
966
967
967
968
969
969
970
971
971
972
973
973
974
975
975
976
977
977
978
979
979
980
981
981
982
983
983
984
985
985
986
987
987
988
989
989
990
991
991
992
993
993
994
995
995
996
997
997
998
999
999
1000
1000
1001
1001
1002
1002
1003
1003
1004
1004
1005
1005
1006
1006
1007
1007
1008
1008
1009
1009
1010
1010
1011
1011
1012
1012
1013
1013
1014
1014
1015
1015
1016
1016
1017
1017
1018
1018
1019
1019
1020
1020
1021
1021
1022
1022
1023
1023
1024
1024
1025
1025
1026
1026
1027
1027
1028
1028
1029
1029
1030
1030
1031
1031
1032
1032
1033
1033
1034
1034
1035
1035
1036
1036
1037
1037
1038
1038
1039
1039
1040
1040
1041
1041
1042
1042
1043
1043
1044
1044
1045
1045
1046
1046
1047
1047
1048
1048
1049
1049
1050
1050
1051
1051
1052
1052
1053
1053
1054
1054
1055
1055
1056
1056
1057
1057
1058
1058
1059
1059
1060
1060
1061
1061
1062
1062
1063
1063
1064
1064
1065
1065
1066
1066
1067
1067
1068
1068
1069
1069
1070
1070
1071
1071
1072
1072
1073
1073
1074
1074
1075
1075
1076
1076
1077
1077
1078
1078
1079
1079
1080
1080
1081
1081
1082
1082
1083
1083
1084
1084
1085
1085
1086
1086
1087
1087
1088
1088
1089
1089
1090
1090
1091
1091
1092
1092
1093
1093
1094
1094
1095
1095
1096
1096
1097
1097
1098
1098
1099
1099
1100
1100
1101
1101
1102
1102
1103
1103
1104
1104
1105
1105
1106
1106
1107
1107
1108
1108
1109
1109
1110
1110
1111
1111
1112
1112
1113
1113
1114
1114
1115
1115
1116
1116
1117
1117
1118
1118
1119
1119
1120
1120
1121
1121
1122
1122
1123
1123
1124
1124
1125
1125
1126
1126
1127
1127
1128
1128
1129
1129
1130
1130
1131
1131
1132
1132
1133
1133
1134
1134
1135
1135
1136
1136
1137
1137
1138
1138
1139
1139
1140
1140
1141
1141
1142
1142
1143
1143
1144
1144
1145
1145
1146
1146
1147
1147
1148
1148
1149
1149
1150
1150
1151
1151
1152
1152
1153
1153
1154
1154
1155
1155
1156
1156
1157
1157
1158
1158
1159
1159
1160
1160
1161
1161
1162
1162
1163
1163
1164
1164
1165
1165
1166
1166
1167
1167
1168
1168
1169
1169
1170
1170
1171
1171
1172
1172
1173
1173
1174
1174
1175
1175
1176
1176
1177
1177
1178
1178
1179
1179
1180
1180
1181
1181
1182
1182
1183
1183
1184
1184
1185
1185
1186
1186
1187
1187
1188
1188
1189
1189
1190
1190
1191
1191
1192
1192
1193
1193
1194
1194
1195
1195
1196
1196
1197
1197
1198
1198
1199
1199
1200
1200
1201
1201
1202
1202
1203
1203
1204
1204
1205
1205
1206
1206
1207
1207
1208
1208
1209
1209
1210
1210
1211
1211
1212
1212
1213
1213
1214
1214
1215
1215
1216
1216
1217
1217
1218
1218
1219
1219
1220
1220
1221
1221
1222
1222
1223
1223
1224
1224
1225
1225
1226
1226
1227
1227
1228
1228
1229
1229
1230
1230
1231
1231
1232
1232
1233
1233
1234
1234
1235
1235
1236
1236
1237
1237
1238
1238
1239
1239
1240
1240
1241
1241
1242
1242
1243
1243
1244
1244
1245
1245
1246
1246
1247
1247
1248
1248
1249
1249
1250
1250
1251
1251
1252
1252
1253
1253
1254
1254
1255
1255
1256
1256
1257
1257
1258
1258
1259
1259
1260
1260
1261
1261
1262
1262
1263
1263
1264
1264
1265
1265
1266
1266
1267
1267
1268
1268
1269
1269
1270
1270
1271
1271
1272
1272
1273
1273
1274
1274
1275
1275
1276
1276
1277
1277
1278
1278
1279
1279
1280
1280
1281
1281
1282
1282
1283
1283
1284
1284
1285
1285
1286
1286
1287
1287
1288
1288
1289
1289
1290
1290
1291
1291
1292
1292
1293
1293
1294
1294
1295
1295
1296
1296
1297
1297
1298
1298
1299
1299
1300
1300
1301
1301
1302
1302
1303
1303
1304
1304
1305
1305
1306
1306
1307
1307
1308
1308
1309
1309
1310
1310
1311
1311
1312
1312
1313
1313
1314
1314
1315
1315
1316
1316
1317
1317
1318
1318
1319
1319
1320
1320
1321
1321
1322
1322
1323
1323
1324
1324
1325
1325
1326
1326
1327
1327
1328
1328
1329
1329
1330
1330
1331
1331
1332
1332
1333
1333
1334
1334
1335
1335
1336
1336
1337
1337
1338
1338
1339
1339
1340
1340
1341
1341
1342
1342
1343
1343
1344
1344
1345
1345
1346
1346
1347
1347
1348
1348
1349
1349
1350
1350
1351
1351
1352
1352
1353
1353
1354
1354
1355
1355
1356
1356
1357
1357
1358
1358
1359
1359
1360
1360
1361
1361
1362
1362
1363
1363
1364
1364
1365
1365
1366
1366
1367
1367
1368
1368
1369
1369
1370
1370
1371
1371
1372
1372
1373
1373
1374
1374
1375
1375
1376
1376
1377
1377
1378
1378
1379
1379
1380
1380
1381
1381
1382
1382
1383
1383
1384
1384
1385
1385
1386
1386
1387
1387
1388
1388
1389
1389
1390
1390
1391
1391
1392
1392
1393
1393
1394
1394
1395
1395
1396
1396
1397
1397
1398
1398
1399
1399
1400
1400
1401
1401
1402
1402
1403
1403
1404
1404
1405
1405
1406
1406
1407
1407
1408
1408
1409
1409
1410
1410
1411
1411
1412
1412
1413
1413
1414
1414
1415
1415
1416
1416
1417
1
```

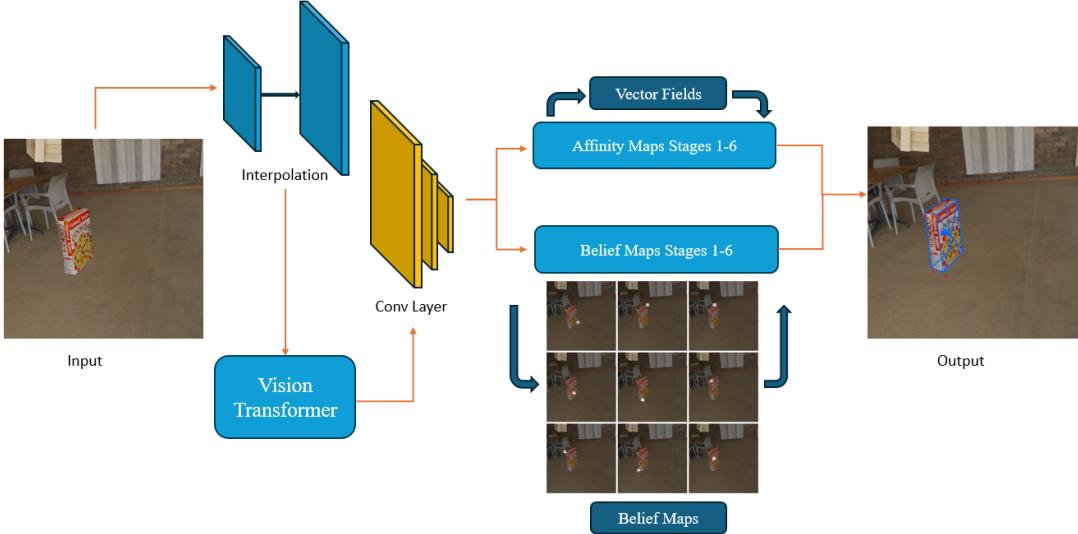


Fig. 3: Enhanced ViT-DOPE Network Architecture

### C. Data Generation

We enhanced the original data generation pipeline [2] using BlenderProc to produce two distinct synthetic RGB datasets, each corresponding to a specific target object: Cookies and Block. The Cookies object is part of the publicly available HOPE dataset [1], while the Block is a newly introduced, custom-designed object. Our pipeline incorporates randomized camera poses, object poses, and 360-degree HDRI backgrounds, while ensuring that these variations remain physically reasonable. These improvements aim to create a more diverse and robust synthetic dataset, helping to mitigate the common sim-to-real domain gap in deep learning applications. The enhanced pipeline consists of four main stages: (1) textured 3D CAD modeling, (2) real-world HDRI background generation, (3) image synthesis, and (4) ground truth annotation pre-processing.

*1) Textured 3D CAD Modeling and Real-World Background Generation:* To obtain a precise 3D textured model of the customized object, we first used SolidWorks to create an accurate geometric model with correct dimensions. Blender was then employed to add textures and enrich visual details, including colors and physical material properties, as shown in Figure 6(b).

For real-world HDRI background generation, we captured raw 360-degree images of the desired physical environments using the Insta360 X3 camera. These images were subsequently pre-processed and converted into HDRI backgrounds using Adobe Photoshop, as illustrated in Figure 6(a).

*2) Image Synthesis:* With all necessary elements prepared, we proceeded to the image synthesis stage. We developed a Python script to randomize the poses of cameras, target objects, and distractors. To emulate typical indoor scenarios encountered in onboard SLAM and manipulation tasks, we assumed that both the camera and the target object remained upright, with randomized yaw angles and small perturbations

in pitch and roll. In contrast, distractor objects were randomized with full degrees of freedom as a form of data augmentation, without adhering to physical stability constraints.

*3) Ground Truth Annotation Pre-Processing:* With the existing pipeline provided by [2], ground truth annotations for each frame were automatically generated. However, when constructing a comprehensive dataset for training and validation, it was necessary to combine synthetic and real images from various sources. In this case, the annotation files (e.g., JSON files) often differed in format and configuration. To streamline data preparation and ensure compatibility with downstream tasks, we developed an additional Python script to pre-process and standardize the ground truth annotations.

### D. Innovative Enhanced Datasets

We augmented the original HOPE dataset and created a new dataset for the customized Block object by generating synthetic domain-randomized (DR) images, referred to as HOPE-Syn&Real and the Synthetic Block Dataset, respectively.

*1) HOPE Data Augmentation (HOPE-Syn&Real Dataset):* We generated additional synthetic data based on the HOPE dataset [1]. The original dataset consists of 28 grocery items, with approximately 300 real images per object. We selected Cookies as the target object for subsequent training tasks. To enrich the existing dataset, we synthesized additional 12,000 domain-randomized (DR) images of this object using the enhanced data generation pipeline developed upon [2], and combined them with the existing real images to form the HOPE-Syn&Real dataset. To verify the quality of the synthesized images, we employed a validation method adapted from the original codebase to visualize the ground truth annotations, as shown in Figure 5.

*2) Synthetic Block Dataset:* In addition to augmenting the HOPE dataset, we created a fully synthetic dataset for our customized Block object using the aforementioned methods

and strategies. This dataset consists of over 19,300 domain-randomized images, with random variations in block poses, instance counts, backgrounds, and distractor objects. Furthermore, as shown in Figure 6(c), lighting conditions and shadows were simulated and rendered to further enhance realism and dataset diversity.

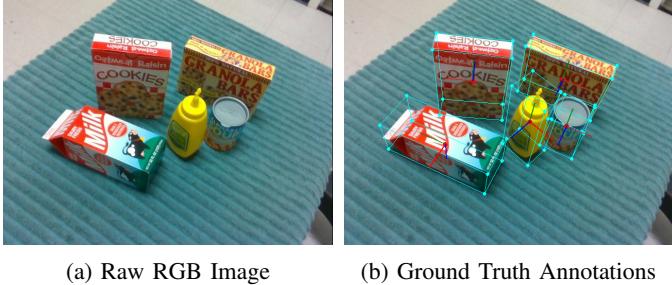


Fig. 4: Sampled HOPE Dataset Real Image and Ground Truths



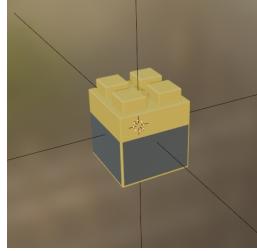
Fig. 5: Sampled Generated Data and Visualized Ground Truth in the HOPE-Syn&Real Dataset. (Left column: generated RGB images, Right column: visualized ground truths)

#### IV. EXPERIMENTS AND RESULTS

Since DOPE is an instance-based network that requires prior knowledge of the target object, each trained model is limited to recognizing a single object. In our experiments, we selected Cookies from the HOPE dataset and Block (a custom object we introduced) as the two target objects. For each object, we trained both the original DOPE model and our modified ViT-DOPE variant. The models were then evaluated based on their final pose estimation results with both visualizations and quantitative metrics, including mean Average Precision (mAP), Average Distance of Model Points (ADD), and predicted keypoints accuracy.



(a) Sampled HDRI Background



(b) 3D Textured Model



(c) Sampled Synthetic Image

Fig. 6: 3D Textured Block Model (Blender), Sampled HDRI Background, and Synthetic Domain Randomized Image in the Synthetic Block Dataset

#### A. Experimental Setup

To facilitate training on the HOPE dataset, we implemented a pre-processing step to convert each object’s pose, which was originally represented as a transformation matrix, into a ground truth belief map. In this dataset, object poses are provided as  $4 \times 4$  transformation matrices that encode the object’s 6D pose relative to the camera, along with the corresponding camera intrinsics. To generate the belief maps, we first reconstructed the 3D bounding boxes of the objects based on their real-world dimensions. These 3D coordinates were then projected onto the 2D image plane using the camera’s intrinsic matrix (as illustrated in Figure 4(b)). After computing the 2D centroids from the projected bounding boxes, we rendered the ground truth belief maps and integrated them into the training pipeline as supervision signals for our model.

#### B. Training

To quantify and compare model performance, we trained a total of four models: one original DOPE and one ViT-DOPE model for each object (Cookies and Block). The HOPE-Syn&Real Dataset and the Synthetic Block Dataset were used to train the Cookies and Block models, respectively. Each dataset was split into training and validation subsets, with the validation sets comprising approximately 5%–7% of the total images. Due to project timeline constraints and the absence of open-source photorealistic data generation scripts in the original DOPE codebase, neither dataset contained photorealistic images. As a result, the Cookies models were trained on both domain-randomized (DR) and real images, whereas the Block models were trained exclusively on DR data.

To ensure a reasonable comparison, all models were trained using identical hyperparameters, optimizers, and learning rate schedules, as detailed in Table I. Specifically, the AdamW optimizer and a Cosine Annealing learning rate schedule were employed, with an initial learning rate of 0.00005. The only difference in the training procedure was the number of epochs: the Cookies models were trained for 200 epochs, whereas the Block models were trained for 400 epochs. The evolution of the learning rate over the course of training is illustrated in Figure 7.

Hyperparameter	Value
Batch Size	64
Initial LR	0.00005
Weight Decay	0.001
Epoch (Cookies)	200
Epoch (Block)	400

TABLE I: Key Hyperparameters and Values

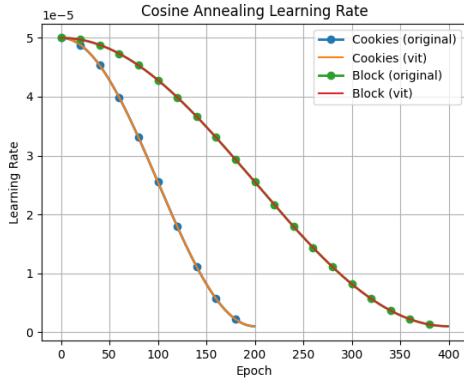


Fig. 7: Learning Rates vs. Epochs

### C. Results and Analysis

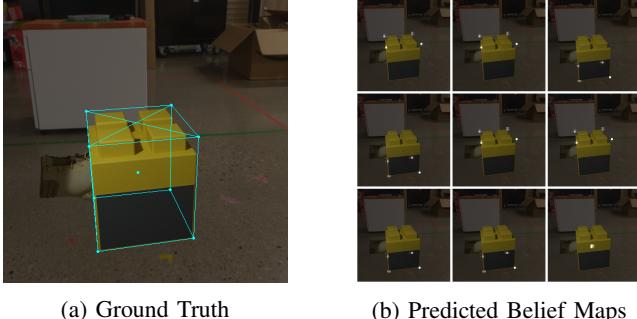


Fig. 8: Ground Truth and Key Point Beliefs of a Block object in a Sampled Frame

The losses and performance metrics of all models converged to stable values with minor fluctuations after training. We define predicted keypoints accuracy as the percentage of keypoints correctly predicted within a specified pixel threshold across a batch of images.

1) *Cookies*: Figure 9 quantifies the performance of the Cookies models on both the training and validation subsets of HOPE-Syn&Real. As shown in Figure 9(a), all losses dropped rapidly during the initial phase of training and gradually stabilized around epoch 125, indicating a reasonable and expected loss convergence trend. In Figure 9(b), the validation accuracy is consistently higher than the training accuracy. This phenomenon suggests either noise in the training set or the presence of strong regularization during training. The former is unlikely, as the images were randomly split into training and validation sets. The latter appears more plausible, as a weight decay rate of 0.001 and additional data augmentation techniques (such as blur and contrast adjustments) were applied during training to mitigate overfitting. Such regularization can lead to higher validation accuracy relative to training accuracy.

When comparing the original DOPE network (using a VGG-19 feature extractor) with our ViT-DOPE architecture, both the training and validation accuracies of ViT-DOPE are approximately 5% higher. This improvement highlights the effectiveness of our architectural enhancements and aligns with the primary goals of this project. Nevertheless, as demonstrated in Figures 9(b) and (d), despite the improvements, the overall estimation performance of both the original DOPE and ViT-DOPE models remains relatively low. Several factors may contribute to this: the batch size was halved due to GPU resource constraints; the dataset lacked photorealistic images; and the total number of synthetic images was limited compared to the original DOPE training process. Specifically, the original DOPE model was trained with approximately 60,000 domain-randomized and 60,000 photorealistic images per object [2], whereas our HOPE-Syn&Real Dataset comprises only about 10.25% of that amount.

Additionally, the absolute values of predicted keypoint accuracies and ADD metrics may have been affected by the relatively strict pixel threshold used for evaluation, where predictions were considered correct only if they fell within approximately 10 pixels of the ground truth. Despite these limitations, the observed trends demonstrate a promising future for DOPE-based architectures. Our results suggest that with stronger feature extractors and further dataset enhancements, significant improvements in 6D pose estimation accuracy are achievable.

2) *Block*: Figure 10 showcases the performance of the Block models on the training and validation splits of the Synthetic Block Dataset. As demonstrated, although the losses remained low after training, the predicted keypoint accuracies plateaued around 20%, and the mAP values stabilized below 0.4, indicating a failure of both the DOPE and ViT-DOPE models for this object. While the absence of photorealistic data may have contributed, the primary cause is attributed to DOPE's core architectural limitations in handling object symmetry [2]. As illustrated in Figure 8, the trained models successfully predicted the nine keypoints of the Block object but failed to generate accurate pose estimations due to the object's geometric and textured symmetry. This also explains the low and oscillating trends observed in the accuracy metrics

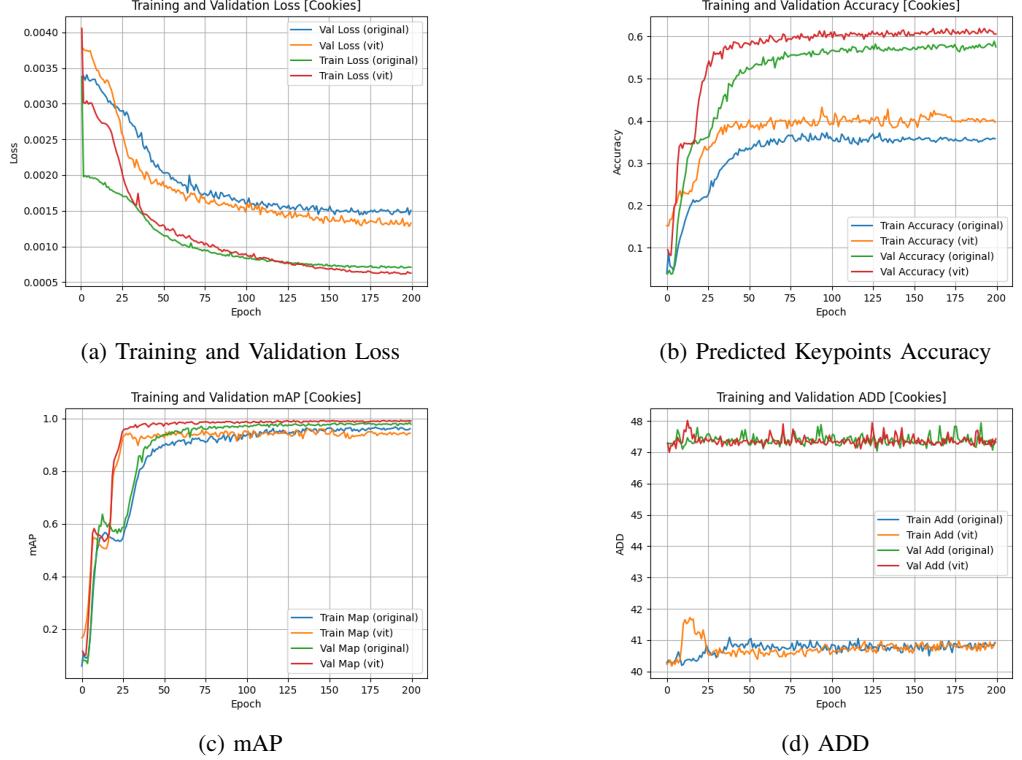


Fig. 9: Cookies Models Performance (Training and Validation)

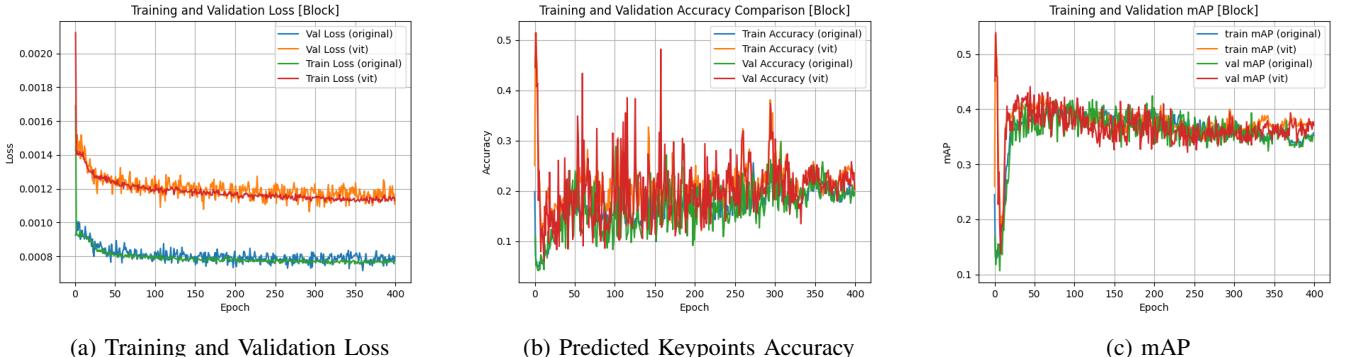


Fig. 10: Block Models Performance (Training and Validation)

in Figure 10.

However, the models' success in locating keypoints of the desired object demonstrated by the belief maps shown in Figure 8(b) highlights the strength of the feature extractors and encourages potential future algorithmic extensions to better handle symmetry.

## V. CONCLUSIONS

Our study successfully extends the original DOPE framework through enhancements in both feature extraction and data generation. By replacing VGG-19 with a vision transformer and increasing the randomization and realism of the datasets, we achieved improvements in object pose estimation

performance. These results demonstrate the potential of ViT for precise feature extraction and highlight the effectiveness of synthetic data in facilitating model training, enabling high model accuracy while significantly reducing the need for manual ground truth annotation. For future research, integrating other advanced feature extractors into the DOPE architecture is a promising direction, as further improvements in estimation performance are likely achievable with more powerful backbones. Additionally, developing algorithmic extensions to better accommodate object symmetry is another important avenue for increasing the adaptability and robustness of the DOPE network.

## REFERENCES

- [1] Y. Lin, J. Tremblay, S. Tyree, P. A. Vela, and S. Birchfield, “Multi-view fusion for multi-level robotic scene understanding,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2021, pp. 6817–6824.
- [2] J. Tremblay, T. To, Y. Xiang, D. Fox, and S. Birchfield, “Deep object pose estimation for semantic robotic grasping of household objects,” *arXiv preprint arXiv:1809.10790*, 2018.
- [3] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, “Attention is all you need,” *Advances in neural information processing systems*, vol. 30, 2017.
- [4] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly *et al.*, “An image is worth 16x16 words: Transformers for image recognition at scale,” *arXiv preprint arXiv:2010.11929*, 2020.
- [5] Y. Xiang, T. Schmidt, V. Narayanan, and D. Fox, “Posecnn: A convolutional neural network for 6d object pose estimation in cluttered scenes,” in *Proceedings of Robotics: Science and Systems (RSS)*, Pittsburgh, Pennsylvania, USA, 2018.
- [6] Y. Wang, Y. Xiang, D. Xu, T. Zhang, T. Khurana, Y. Xiao, A. Deng, and D. Fox, “Densefusion: 6d object pose estimation by iterative dense fusion,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 3343–3352.