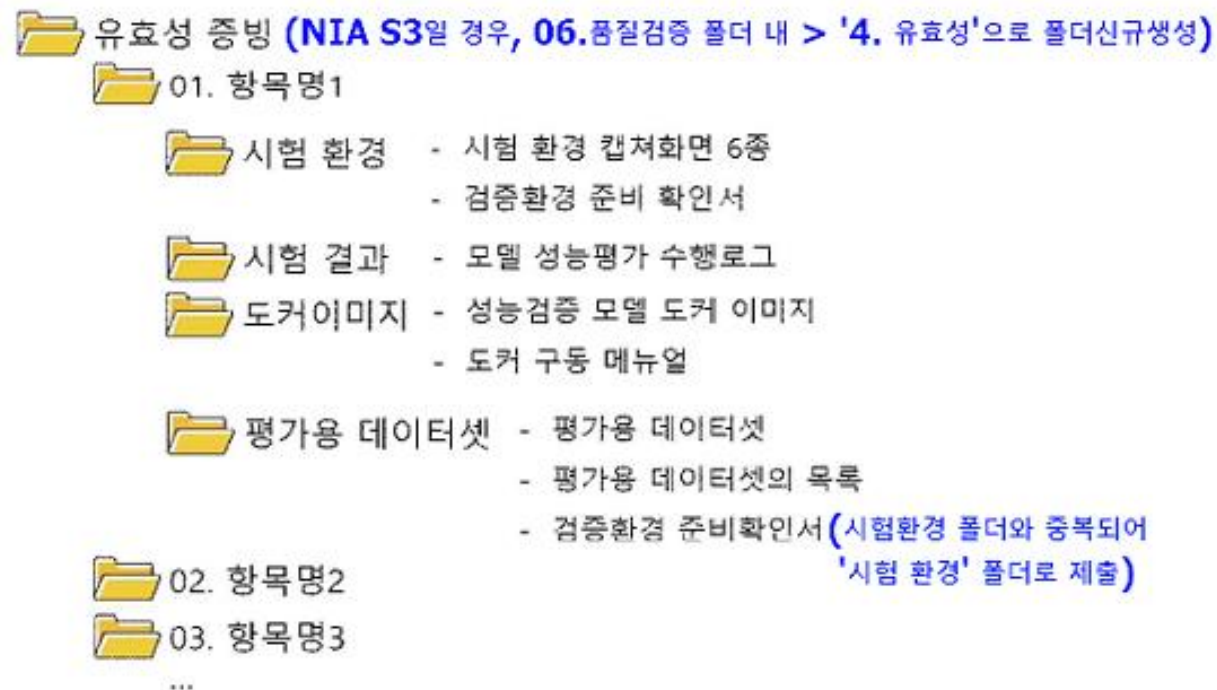


# 객체 3D 위치/자세 인식 모델 실행 메뉴얼

투명/대용량 객체 3D 데이터 구축 사업

2022.12.31

- S3 Browser
- object3d/06.품질검증/4.유효성/
  - 07.투명객체/
  - 08.대용량/
- experimental\_enviroment/ (시험환경)
  - 시험 환경 캡처 6종 (CPU, GPU 등)
  - 검증 환경 준비 확인서
- experimental\_results/ (시험결과)
  - 성능 분석 및 로그 파일 (.txt, .csv)
- docker\_images/ (도커이미지)
  - 성능 검증 모델 도커 이미지
  - 도커 구동 메뉴얼
- test\_datasets/ (평가용 데이터셋)
  - 평가용 데이터셋



- S3 Browser를 활용하여 도커 이미지를 다운 받아 로드하는 경우,

> docker load -i docker\_images/nia-ssp.tar

```
seongheum@vip-02: ~/Workspace/NIA-3DObj/nia-ssp-docker
seongheum@vip-02:~/Workspace/NIA-3DObj/nia-ssp-docker$ docker images
REPOSITORY TAG IMAGE ID CREATED SIZE
seongheum@vip-02:~/Workspace/NIA-3DObj/nia-ssp-docker$ docker ps -a
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
seongheum@vip-02:~/Workspace/NIA-3DObj/nia-ssp-docker$ ls docker_images/
nia-ssp.tar
seongheum@vip-02:~/Workspace/NIA-3DObj/nia-ssp-docker$ docker load -i docker_images/nia-ssp.tar
```

- Git에서 관리되는 소스코드를 build하여 도커 이미지를 생성하는 경우,

> git clone <https://github.com/seongheum-ssu/nia-ssp>

> ./build.sh

```
seongheum@vip-02: ~/Workspace/NIA-3DObj/nia-ssp-docker
seongheum@vip-02:~/Workspace/NIA-3DObj/nia-ssp-docker$ docker images
REPOSITORY TAG IMAGE ID CREATED SIZE
seongheum@vip-02:~/Workspace/NIA-3DObj/nia-ssp-docker$ ls docker_images/cfg/
Dockerfile environment.yaml
seongheum@vip-02:~/Workspace/NIA-3DObj/nia-ssp-docker$ ls docker_images/src
cfg.py darknet.py dataset.py image.py MeshPly.py prepare.py project.py region_loss.py train.py utils.py valid.py
seongheum@vip-02:~/Workspace/NIA-3DObj/nia-ssp-docker$ ./build.sh
Sending build context to Docker daemon 3.386MB
Step 1/14 : FROM nvidia/cuda:11.1-devel-ubuntu18.04
11.1-devel-ubuntu18.04: Pulling from nvidia/cuda
f22ccc0b8772: Downloading [=====>] 6.695MB/26.71MB
3cf8fb62ba5f: Verifying Checksum
e80c964ece6a: Download complete
8a451ac89a87: Downloading [=====>] 6.036MB/7.232MB
c563160b1f64: Downloading [=>] 229.4kB/10.79MB
596a46902202: Waiting
aa0805983180: Pulling fs layer
5718c3da35a0: Waiting
```

```
Step 13/14 : COPY docker_images/src/*.py /ssp/
---> d0c18eda4d41
Step 14/14 : CMD python valid.py --datacfg data/sample/sample.data
up.weights
---> Running in 87f629f809f1
Removing intermediate container 87f629f809f1
---> af39f2069b9a
Successfully built af39f2069b9a
Successfully tagged nia-ssp:1.0
seongheum@vip-02:~/Workspace/NIA-3DObj/nia-ssp-docker$
```

```
seongheum@vip-02: ~/Workspace/NIA-3DObj/nia-ssp-docker
seongheum@vip-02:~/Workspace/NIA-3DObj/nia-ssp-docker$ docker images
REPOSITORY TAG IMAGE ID CREATED SIZE
nia-ssp 1.0 af39f2069b9a 14 minutes ago 16.1GB
nvidia/cuda 11.1-devel-ubuntu18.04 2e6eddb41f77 12 months ago 4.74GB
seongheum@vip-02:~/Workspace/NIA-3DObj/nia-ssp-docker$
```

```
seongheum@vip-02: ~/Workspace/NIA-3DObj/nia-ssp-docker
seongheum@vip-02:~/Workspace/NIA-3DObj/nia-ssp-docker$ ls test_datasets/
030102.zip 050202.zip 050311.zip 060108.zip 060211.zip 070308.zip 070605.zip 070611.zip 070708.zip 070911.zip 100205.zip
050110.zip 050210.zip 050312.zip 060201.zip 060302.zip 070403.zip 070608.zip 070702.zip 070710.zip 090105.zip 100211.zip
050201.zip 050305.zip 060106.zip 060207.zip 070205.zip 070409.zip 070610.zip 070704.zip 070902.zip 090206.zip
seongheum@vip-02:~/Workspace/NIA-3DObj/nia-ssp-docker$ cat run.sh
#!/bin/bash

TESTCASE=("030102" "050110" "050201" "050202" "050210" "050305" "050311" "050312" "060106" "060108" "060201" "060207" "060211" "060302" "070205" "070308" "070403" "070409" "070605" "070608" "070610" "070611" "070702" "070704" "070708" "070710" "070902" "070911" "090105" "090206" "100205" "100211")

if [ $1 == "all" ]; then
    # RUN ALL
    for i in ${TESTCASE[@]}
    do
        eval "./prepare.sh ${i}"
        eval "./test.sh ${i}"
    done
else
    for i in ${TESTCASE[@]}
    do
        if [ $1 -eq "${i}" ]; then
            eval "./prepare.sh ${i}"
            eval "./test.sh ${i}"
        fi
    done
fi

seongheum@vip-02:~/Workspace/NIA-3DObj/nia-ssp-docker$ ./run.sh all
```

테스트 목록 (21.12.31)

전체 실행

선택 실행

- S3 Browser를 활용하여 평가용 데이터셋 다운로드. 테스트 목록 확인.
  - 테스트 목록 전체 실행, 객체 ID별 실행
- > ./run.sh all
- > ./run.sh 100211

## 평가 수행

```

inflating: masks/100211_001199.png
inflating: masks/100211_001200.png
creating: models/
inflating: models/model.weights
inflating: models/yolo-pose.cfg
inflating: test.txt
inflating: train.txt
docker run --name=NIA-SSP --runtime=nvidia --rm -v /home/seongheum/Workspace/NIA-3DObj/nia-ssp-docker/test_datasets:ssp/data -v /home/seongheum/Workspace/NIA-3DObj/nia-ssp-docker/experimental_results:ssp/experimental_results nia-ssp:1.0 python valid.py --datacfg data/100211/100211.data --modelcfg data/100211/models/yolo-pose.cfg --weightfile data/100211/models/model.weights
2022-01-03 10:29:18 Results of 100211 (2022-01-03 10:29:18.818820)
2022-01-03 10:29:18 Mean Err. (Pixel Dist.) = 7.78 pix.
2022-01-03 10:29:18 Acc. using 10 px 2D Projection = 78.33%
2022-01-03 10:29:18 Acc. using Intersection Of Union (IoU) = 100.00%
seongheum@vip-02:~/Workspace/NIA-3DObj/nia-ssp-docker$
    
```

#	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	AA	AB	AC	AD	AE	AF	AG	AH	AI	AJ	AK	AL	AM	AN	AO
1	Data ID	x0-GT	y0-GT	x1-GT	y1-GT	x2-GT	y2-GT	x3-GT	y3-GT	x4-GT	y4-GT	x5-GT	y5-GT	x6-GT	y6-GT	x7-GT	y7-GT	x8-GT	y8-GT	x0-predict	y0-predict	x1-predict	y1-predict	x2-predict	y2-predict	x3-predict	y3-predict	x4-predict	y4-predict	x5-predict	y5-predict	x6-predict	y6-predict	x7-predict	y7-predict	x8-predict	y8-predict	pixel error	2D projec	IoU	IoU score
2	100211_C	734	353	793	445	790	340	660	430	656	329	812	365	807	273	686	355	683	262	737.3	352.1	783.8	442.8	780.9	340.1	662.1	425	659.8	329.8	806.9	361.6	805	271.8	688.6	348.8	691.6	261.7	3.85	True	0.95	True
3	100211_C	679	352	643	434	626	343	620	366	611	288	747	413	734	324	724	346	713	271	679.5	350.6	634.3	428	640.4	346.7	611	353.1	616	293.2	750.3	402.3	728.2	328.6	719	332.4	703.2	277.4	7.31	True	0.9	True
4	100211_C	679	289	641	290	638	204	755	295	757	208	605	365	601	263	741	374	739	271	681.9	292	642.9	287	647.2	204.2	756	308.4	754.9	219.6	603.3	356.7	610.5	253.3	730.6	384.5	727.8	276.5	5.84	True	0.93	True
5	100211_C	676	287	687	283	680	194	756	319	752	228	603	340	597	245	677	380	674	280	663.6	282.8	644.7	284.5	646	197.4	742.8	310.9	743.6	217.7	587.2	337.6	586.4	238.5	693.4	369.4	689.8	264	14.75	False	0.88	True
6	100211_C	664	284	582	389	578	252	581	312	578	180	750	389	746	252	749	312	741	180	663.6	284.1	586.9	390.8	576.2	241.6	598.8	310.1	593	175.8	748.9	389.9	740.1	240.7	754.2	308.9	747.1	171.2	7.3	True	0.95	True
7	100211_C	701	289	703	288	703	200	794	332	792	231	614	328	608	233	704	379	701	266	696	285.5	687.4	289.5	688.3	194.1	773.3	323.2	776.4	221.4	615.4	338.2	615.3	233.5	708.9	377.2	707.8	265.3	6.95	True	0.91	True
8	100211_C	669	327	612	412	607	324	621	311	620	242	725	411	722	324	732	311	731	243	677	332.2	603.9	404.6	609.6	317.7	627.5	316.1	629.1	252.3	722.4	408.8	709.4	322.4	735	317.4	724.4	257.9	5.25	True	0.9	True
9	100211_C	676	314	750	337	743	269	688	398	683	323	668	290	665	230	604	342	602	277	680	314	748.5	336.2	740	265.2	688.5	389	687.8	305.9	676.3	304.4	673	240.1	613.1	350.6	616.8	275.4	7.78	True	0.91	True
10	100211_C	707	326	678	428	674	335	626	333	625	263	789	378	789	290	731	293	732	225	703.8	330.9	676	422	673.7	327.7	626.4	350.3	626.5	270.7	787.5	380.1	775.8	294	728.9	314.4	722.6	242	9.69	True	0.92	True
11	100211_C	643	361	639	353	636	283	711	378	706	310	580	401	575	328	652	439	647	358	639.9	357.5	629.3	340.3	625.2	275.7	709.4	374.9	707.3	305.7	574.7	400.5	570	327.5	662.1	441.3	655.4	365.5	4.91	True	0.93	True
12	100211_C	745	305	813	322	816	234	790	396	792	302	707	215	675	372	677	279	746	279	746	301.9	815.1	317.2	816.5	236.6	784.5	387.4	788.4	290.4	710.1	295.2	716.3	217.4	670.6	359.4	675.1	267.2	7	True	0.93	True
13	100211_C	648	300	584	306	582	218	709	304	700	216	586	385	584	296	714	383	708	295	642	294.3	592.9	292.2	596.9	208.8	710.1	303.9	707.5	217.9	570.3	366.5	573.6	267	699.6	381.3	696.2	277.9	12	False	0.89	True
14	100211_C	658	359	708	337	683	282	722	404	699	344	602	371	594	310	623	437	610	372	654.8	355	688	340.3	679.2	276.5	721	404.3	718.1	332.1	592.6	368.5	584.4	299.6	623.9	434.7	618.2	360.4	8.06	True	0.93	True

## 로그 파일

- 테스트 케이스별로 다음의 정보를 확인
  - Timestamp (서울 표준시 기록), 실행 명령어 (명령어 history),
  - 개별 결과값, 최종 결과값, 최종 결과 계산 시 사용한 값 (reprojection error, IoU score)
- reprojection 기반 정확도, IoU 기반 정확도로 위치/자세 추정 결과에 대해 성능 분석
  - 로그 파일에는 GT에서 제공된 큐브 꼭지점 좌표 위치와 모델에서 예측한 위치를 비교
  - 테스트 영상 별로 평균 픽셀 오차와 IoU 스코어를 비교하여 목표치 도달 여부 판단

```
seongheum@vip-02:~/Workspace/NIA-3DObj/nia-ssp-docker$ ls experimental_results/
030102.csv  050201.txt  050305.csv  050312.txt  060201.csv  060211.txt  070308.csv  070409.txt  070610.csv  070702.txt  070710.csv  070911.txt  100205.csv
030102.txt  050202.csv  050305.txt  060106.csv  060201.txt  060302.csv  070308.txt  070605.csv  070610.txt  070704.csv  070710.txt  090105.csv  100205.txt
050110.csv  050202.txt  050311.csv  060106.txt  060207.csv  060302.txt  070403.csv  070605.txt  070611.csv  070704.txt  070902.csv  090105.txt  100211.csv
050110.txt  050210.csv  050311.txt  060108.csv  060207.txt  070205.csv  070403.txt  070608.csv  070611.txt  070708.csv  070902.txt  090206.csv  100211.txt
050201.csv  050210.txt  050312.csv  060108.txt  060211.csv  070205.txt  070409.csv  070608.txt  070702.csv  070708.txt  070911.csv  090206.txt
seongheum@vip-02:~/Workspace/NIA-3DObj/nia-ssp-docker$
```

```
seongheum@vip-02:~/Workspace/NIA-3DObj/nia-ssp-docker$ ./prepare.sh report
seongheum@vip-02:~/Workspace/NIA-3DObj/nia-ssp-docker$ cat report.txt
report
030102 8.62 83.33 100.00
050110 3.71 100.00 100.00
050201 4.11 99.58 100.00
050202 6.42 91.25 100.00
050210 3.35 100.00 100.00
050305 11.91 64.58 100.00
050311 30.64 42.44 87.82
050312 5.24 96.49 100.00
060106 4.03 100.00 100.00
060108 7.80 78.75 100.00
060201 4.55 97.08 100.00
060207 6.41 95.00 99.58
060211 5.92 90.83 100.00
060302 6.23 93.33 100.00
070205 10.72 70.83 100.00
070308 6.70 89.17 100.00
070403 7.12 92.80 98.31
070409 2.46 100.00 100.00
070605 6.52 87.50 100.00
070608 7.26 91.67 100.00
070610 4.27 97.90 100.00
070611 7.05 84.58 100.00
070702 2.12 99.58 100.00
070704 4.02 97.08 100.00
070708 10.28 81.67 100.00
070710 4.82 97.50 100.00
070902 10.83 68.75 100.00
070911 36.99 89.50 95.38
090105 2.56 100.00 100.00
090206 3.70 99.17 100.00
100205 4.24 97.48 100.00
100211 7.78 78.33 100.00
seongheum@vip-02:~/Workspace/NIA-3DObj/nia-ssp-docker$
```

## 학습 유효성 검증

030102	83.33	100
050110	100	100
050201	99.58	100
050202	91.25	100
050210	100	100
050305	64.58	100
050311	42.44	87.82
050312	96.49	100
060106	100	100
060108	78.75	100
060201	97.08	100

060207	95	99.58
060211	90.83	100
060302	93.33	100
070205	70.83	100
070308	89.17	100
070403	92.8	98.31
070409	100	100
070605	87.5	100
070608	91.67	100
070610	97.9	100
070611	84.58	100

070702	99.58	100
070704	97.08	100
070708	81.67	100
070710	97.5	100
070902	68.75	100
070911	89.5	95.38
090105	100	100
090206	99.17	100
100205	97.48	100
100211	78.33	100
평균	<b>89.25531</b>	<b>99.40906</b>

> prepare.sh report → 평가용 데이터셋 목록의 최종 결과 값을 report.txt에 정리

- 2022.01.03: 업데이트 된 테스트 목록으로 유효성 검증 진행