

# Hailo DFC Guide

## with Docker

© WITHUS 2023, All Rights Reserved, V1.2



## 목차

<b>I</b>	<b>개요</b>	<b>4</b>
1.	문서 목적	4
2.	History	5
<b>II</b>	<b>준비</b>	<b>6</b>
1.	구성	6
1.1	시스템 구성	6
1.2	파일 구성	6
2.	Docker 설치	6
2.1	Apt repository 기반 설치	6
2.2	Docker 패키지 설치	7
2.3	Docker 테스트	7
3.	Docker 설정	8
3.1	Docker 권한 부여	8
4.	Hailo sw suite docker 준비	9
4.1	다운로드	9
4.2	압축해제	9
4.3	실행	9
4.4	참고	9
<b>III</b>	<b>DFC</b>	<b>10</b>
1.	사전 준비	10
1.1	다운로드	10
1.2	복사	10
1.3	압축해제	10
1.4	공유 폴더 권한 변경	10
1.5	디렉토리 확인 및 수정	11
1.6	Docker실행 및 재접속	12
1.7	Jupyter노트북 파일 복사	12
2.	실행	13

2.1 Jupyter Notebook 실행 ..... 13

2.2 DFC 진행..... 13

# I 개요

---

## 1. 문서 목적

본 문서는 Docker 환경에서 YOLOv5 기반 ONNX를 HEF파일로 변환하기 위한 가이드입니다.

### 2. History

Version	날짜	내용
1.0	2023.09.15.	Initial Release
1.1	2024.05.30	Jupyter 내용 추가
1.2	2024.06.25	- GPU 내용 삭제 - Docker Shared Folder 관련 내용 업데이트 - 가이드파일 변경

## II 준비

### 1. 구성

#### 1.1 시스템 구성

- CPU: Intel or amd 64bit Processor
- Memory: 16GB 이상
- OS: Ubuntu 20.04 (본 문서에서는 해당 OS를 기준으로 설명합니다.)

#### 1.2 파일 구성

- Hailo SW Suite docker(약 8.7GB):  
[https://drive.google.com/file/d/180Tw4spRo7D81ommwHnWZhCeiQaDRj/view?usp=drive\\_link](https://drive.google.com/file/d/180Tw4spRo7D81ommwHnWZhCeiQaDRj/view?usp=drive_link)
- withus\_dfc\_yolov5m.zip 파일 (약 100MB):  
<https://drive.google.com/file/d/1kjpPgYBbX7ZMmPkqv40-yGj6VKteMTMS/view?usp=sharing>

### 2. Docker 설치

#### 2.1 Apt repository 기반 설치

- Add Docker's official GPG key:

```
sudo apt-get update
```

- Add Key

```
sudo apt-get install ca-certificates curl gnupg
sudo install -m 0755 -d /etc/apt/keyrings
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o /etc/apt/keyrings/docker.gpg
sudo chmod a+r /etc/apt/keyrings/docker.gpg
```

- Add the repository to Apt sources:

```
echo "deb [arch="$(dpkg --print-architecture)" \
signed-by=/etc/apt/keyrings/docker.gpg] https://download.docker.com/linux/ubuntu \
"$(. /etc/os-release && echo "$VERSION_CODENAME)" stable" | \
sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
```

- Update

```
sudo apt-get update
```

### 2.2 Docker 패키지 설치

```
sudo apt-get install docker-ce docker-ce-cli containerd.io docker-buildx-plugin \  
docker-compose-plugin
```

### 2.3 Docker 테스트

```
sudo docker images
```

### 3. Docker 설정

#### 3.1 Docker 권한 부여

```
sudo usermod -aG docker ${USER}
```

##### 3.1.1 Docker.sock permission 오류가 나는 경우

```
sudo chmod 666 /var/run/docker.sock
```

```
# 권한 변경 후 로그아웃 및 재로그인 필요
```



### 4. Hailo sw suite docker 준비

#### 4.1 다운로드

준비 섹션에서 다운로드하지 않은 경우 다운로드합니다.

- Hailo SW Suite docker(약 8.7GB):

[https://drive.google.com/file/d/180Tw4spRo7D81ommwohHnWZhCeiQaDRj/view?usp=drive\\_link](https://drive.google.com/file/d/180Tw4spRo7D81ommwohHnWZhCeiQaDRj/view?usp=drive_link)

#### 4.2 압축해제

```
unzip hailo_sw_suite_2023-07.1_docker.zip
```

#### 4.3 실행

```
./hailo_sw_suite_docker_run.sh
```

#### 4.4 참고

- Docker container 내부에서 빠져나가기

```
exit
```

- 실행중인 docker container 확인

```
docker ps
```

- 재접속

```
./hailo_sw_suite_docker_run.sh -resume
```

- Docker Container 외부에서 실행 중인 컨테이너 종료

```
docker stop hailo_sw_suite_2022-07
```

- docker image 목록확인

```
sudo docker images
```

## III DFC

### 1. 사전 준비

#### 1.1 다운로드

- 준비 섹션에서 다운로드하지 않은 경우 다운로드합니다.
- withus\_dfc\_yolov5m.zip:  
<https://drive.google.com/file/d/1kjpGyYBbX7ZMmPkqv40-yGJ6VKteMTMS/view?usp=sharing>

#### 1.2 복사

- 제공된 Docker shell script를 통해 실행하면, 실행위치에 shared\_with\_docker 폴더가 생성됩니다.
- Shared\_with\_docker 폴더로 다운로드한 withus\_dfc\_yolov5m.zip을 복사합니다.
- 아래 예시의 docker root경로는 "~/Work/dfc"입니다.

#### 1.3 압축해제

- 복사한 파일의 압축을 해제합니다.

```
withus@withus-Server:~/Work/dfc/shared_with_docker$ unzip withus_dfc_yolov5m.zip
```

#### 1.4 공유 폴더 권한 변경

```
withus@withus-Server:~/Work/dfc$ sudo chmod -R 777 shared_with_docker
withus@withus-Server:~/Work/dfc$ ll shared_with_docker/
합계 184556
drwxrwxrwx 7 withus withus    4096  6월 25 15:46 ./
drwxrwxr-x 3 withus withus    4096  6월 25 11:19 ../
-rwxrwxrwx 1 withus withus 1037234  6월 24 17:45 DFC-yolov5m.ipynb*
drwxrwxrwx 3 withus withus    4096  6월 25 14:58 dataset/
drwxrwxrwx 2 10642 10600     4096  6월 25 11:19 doc/
drwxrwxrwx 2 withus withus    4096  6월 25 15:02 har/
drwxrwxrwx 2 withus withus    4096  6월 25 14:59 hef/
drwxrwxrwx 2 withus withus    4096  6월 25 14:59 svg/
-rwxrwxrwx 1 withus withus 103887921  6월 25 15:45 withus_dfc_yolov5m.zip*
```

```
-rwxrwxrwx 1 withus withus 84019785 6월 24 17:44 yolov5m_coco_10class.onnx*
```

### 1.5 디렉토리 확인 및 수정

- DFC-yolov5m.ipynb

dfc를 위한 Jupyter 파일입니다. 커스텀 모델 학습 시, 경로 / 이미지 포맷 / 모델이름 등의 내용 수정이 필요합니다.

- Dataset 폴더

calibration용 이미지 폴더입니다. 기본으로 notebook파일에 “coco\_10class”가 설정되어있으며, 커스텀모델의 경우, Calibration 이미지를 폴더 생성 후 저장합니다. 이 후 notebook파일에 해당경로를 설정합니다.

- har, hef, svg 폴더: DFC 산출물 저장 폴더입니다.
- 변환할 onnx파일: 변환할 onnx파일을 준비합니다.

### 1.6 Docker실행 및 재접속

- 경로 확인

```
withus@withus-Server:~/Work/dfc$ ll
합계 25659208

drwxrwxr-x  3 withus withus    4096  6월 25 11:19 ./
drwxrwxr-x 19 withus withus    4096  6월 25 10:51 ../
-rw-----  1 withus withus 16981737984  7월 16  2023 hailo_sw_suite_2023-07.1.tar
-rwxrw-r--  1 withus withus  9293247668   6월 25  10:59 hailo_sw_suite_2023-07.1_docker.zip*
-rwxr-xr-x  1 withus withus   12944  6월 25 11:18 hailo_sw_suite_docker_run.sh*
drwxrwxrwx  7 withus withus    4096  6월 25 15:02 shared_with_docker/
```

- 실행

```
withus@withus-Server:~/Work/dfc$ ./hailo_sw_suite_docker_run.sh
```

- 재접속

```
withus@withus-Server:~/Work/dfc$ ./hailo_sw_suite_docker_run.sh -resume
```

### 1.7 Jupyter노트북 파일 복사

- 실행된 Docker Container 내에서 Shared\_with\_docker폴더의 Jupyter notebook 파일을 hailo tutorial 환경의 워크스페이스로 복사합니다.

```
(hailo_virtualenv) hailo@withus-Server:/local/workspace$ cp ../shared_with_docker/DFC-yolov5m.ipynb ./hailo_virtualenv/lib/python3.8/site-packages/hailo_tutorials/notebooks/
```

# Hailo DFC Guide

## 2. 실행

### 2.1 Jupyter Notebook 실행

- Docker command line에서 jupyter notebook 실행합니다. IP는 루프백 IP 또는 설정된 IP를 입력합니다. (ifconfig 명령을 통해 확인하실 수 있습니다.)

```
(hailo_virtualenv) hailo@withus-Server:/local/workspace$ hailo tutorial --ip=0.0.0.0
```

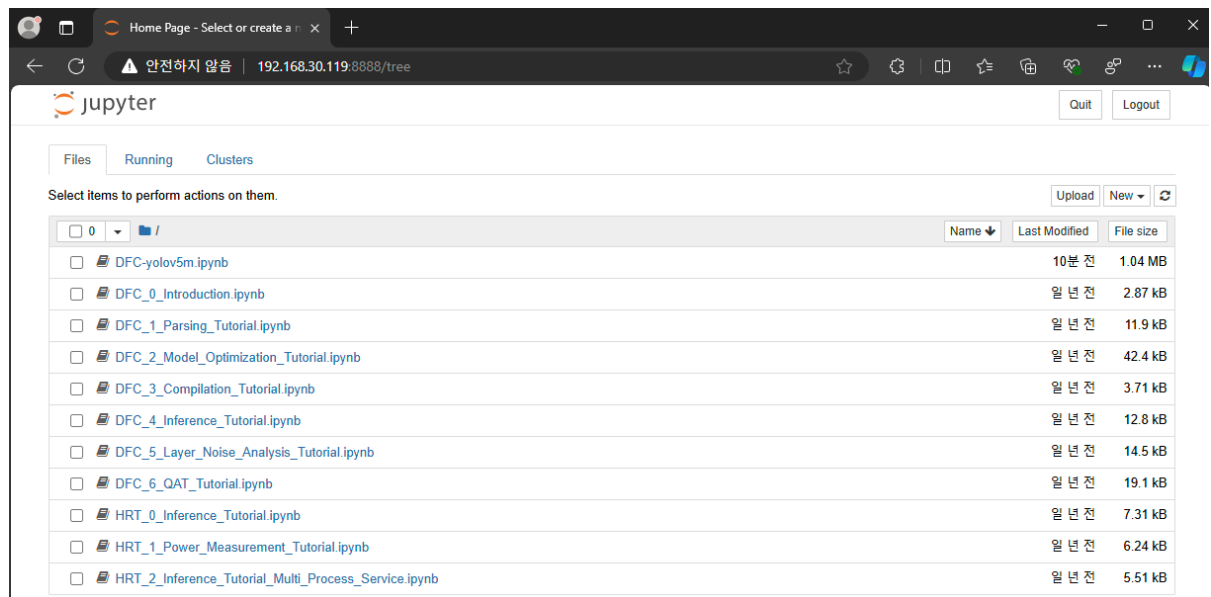
- 아래와 같이 출력됩니다.

```
(hailo_virtualenv) hailo@withus-Server:/local/workspace$ hailo tutorial --ip=192.168.30.119
[Info] First time Hailo Dataflow Compiler is being used. Checking system requirements... (this might take a few seconds)
[Info] No GPU connected.
[15:04:07.944 NotebookApp] Writing notebook server cookie secret to /home/hailo/.local/share/jupyter/runtime/notebook_cookie_secret
[15:04:08.650 NotebookApp] Serving notebooks from local directory: /local/workspace/hailo_virtualenv/lib/python3.8/site-packages/hailo_tutorials/notebooks
[15:04:08.650 NotebookApp] Jupyter Notebook 6.5.2 is running at:
[15:04:08.650 NotebookApp] http://192.168.30.119:8888/?token=e6b23d92d16e7f274b3decda1c342d722b1facc180b82735
[15:04:08.650 NotebookApp] or http://127.0.0.1:8888/?token=e6b23d92d16e7f274b3decda1c342d722b1facc180b82735
[15:04:08.650 NotebookApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation).
[C 15:04:08.957 NotebookApp]

To access the notebook, open this file in a browser:
file:///home/hailo/.local/share/jupyter/runtime/nbserver-124-open.html
Or copy and paste one of these URLs:
http://192.168.30.119:8888/?token=e6b23d92d16e7f274b3decda1c342d722b1facc180b82735
or http://127.0.0.1:8888/?token=e6b23d92d16e7f274b3decda1c342d722b1facc180b82735
```

### 2.2 DFC 진행

- 실행 시 출력된 웹경로를 웹브라우저에 copy/paste 후 접속합니다. 접속 후 아래 화면을 확인하실 수 있습니다.



- DFC-yolov5m.ipynb 파일을 클릭하여 실행합니다.
- 이 후 파일 내용에 따라 진행합니다. PC 사양에 따라 30분이상 걸릴 수 있습니다.

## Hailo DFC Guide

- 완료 시 아래 내용을 확인하실 수 있습니다.

```
In [11]: hef = runner.compile()

file_name = workpath + 'hef/' + onnx_model_name + '.hef'
with open(file_name, 'wb') as f:
    f.write(hef)
```

[info]	cluster_0	68.8%	71.9%	50.8%
[info]	cluster_1	62.5%	68.8%	51.6%
[info]	cluster_2	56.3%	60.9%	46.1%
[info]	cluster_3	12.5%	18.8%	12.5%
[info]	cluster_4	100%	73.4%	58.6%
[info]	cluster_5	25%	14.1%	16.4%
[info]	cluster_6	68.8%	79.7%	58.6%
[info]	cluster_7	62.5%	62.5%	48.4%
[info]	Total	57%	56.3%	42.9%

```
[info] Successful Mapping (allocation time: 14m 16s)
[info] Compiling context_0...
[info] Compiling context_1...
[info] Compiling context_2...
[info] Compiling context_3...
[info] Building HEF...
[info] Successful Compilation (compilation time: 26s)
```

- 이 후 “Shared\_with\_docker” 폴더 내 hef 폴더에서 생성된 hef파일을 확인하실 수 있습니다.

```
withus@withus-Server:~/Work/dfc/shared_with_docker/hef$ ll
한 계 22616
drwxrwxrwx 2 withus withus 4096 6월 25 16:20 ./
drwxrwxrwx 7 withus withus 4096 6월 25 15:46 ../
-rw-rw-rw- 1 10642 10600 23148963 6월 25 16:20 yolov5m_coco_10class.hef
```