Reference Materials

Hi everyone,

I have put together some materials that might help you guys with assignment 2. The set up for KVS(mac,msys,docker) given at the end, was posted in piazza by students from last semester. If anyone finds out easier/better ways to set up KVS, do share it in piazza for your classmates. Hope these materials are helpful.

**References for Kinesis**

[Facial Recognition with a Raspberry Pi and Kinesis Video Streams — Part 1](https://medium.com/@matt.collins/facial-recognition-with-a-raspberry-pi-and-kinesis-video-streams-part-1-662f0bec5488)

[Facial Recognition with a Raspberry Pi and Kinesis Video Streams — Part 2](https://medium.com/@matt.collins/facial-recognition-with-a-raspberry-pi-and-kinesis-video-streams-part-2-9c9a631e8c24)

[AWS Kinesis Data Streams — a Tiny CLI Demo](https://medium.com/@matt.collins/facial-recognition-with-a-raspberry-pi-and-kinesis-video-streams-part-2-9c9a631e8c24)

[Kinesis Streams Demo](https://www.youtube.com/watch?v=9RIBgb4vxug)

[Display a video stream from a local camera](https://www.youtube.com/watch?v=6zpcNxrvtSc)

[AWS Kinesis: Introduction and How to Upload Data](https://www.youtube.com/watch?v=k38tZ-M9o50)

[AWS Kinesis: How to Connect a Data Stream to Lambda Function](https://www.youtube.com/watch?v=nlxk8hFkQOg)

[AWS Kinesis Stream firehose Console Overview | AWS kinesis firehose | AWS Tutorial for beginners](https://www.youtube.com/watch?v=W1uqtxwIipQ)

Create a Kinesis Video Stream:

<https://docs.aws.amazon.com/kinesisvideostreams/latest/dg/gs-createstream.html>

Send Data to a Kinesis Video Stream

<https://docs.aws.amazon.com/kinesisvideostreams/latest/dg/gs-send-data.html>

**References for Rekognition**

The Rekognition part can be done by the AWS SDK.

For example, if you are using Python (boto3 SDK), then first you can define a client like:

rek = boto3.client('rekognition')

Then detect the labels of current image:

labels = rek.detect\_labels(

Image={

'XXObject': {

}

},

MaxLabels=XXX

)

Here are some references that might help:

[How to build a serverless clone of Imgur using Amazon Rekognition and DynamoDB](https://read.acloud.guru/building-an-imgur-clone-part-2-image-rekognition-and-a-dynamodb-backend-abc9af300123)

[Response: Racial and Gender bias in Amazon Rekognition — Commercial AI System for Analyzing Faces.](https://medium.com/@Joy.Buolamwini/response-racial-and-gender-bias-in-amazon-rekognition-commercial-ai-system-for-analyzing-faces-a289222eeced)

[A Quick Introduction to AWS Rekognition](https://medium.com/faun/a-quick-introduction-to-aws-rekognition-8257d4777198)

**Retrieving frames from Kinesis Stream**

[https://docs.aws.amazon.com/de\_de/cli/latest/reference/kinesis-video-archived-media/get-media-for-fragment-list.html](https://urldefense.proofpoint.com/v2/url?u=https-3A__docs.aws.amazon.com_de-5Fde_cli_latest_reference_kinesis-2Dvideo-2Darchived-2Dmedia_get-2Dmedia-2Dfor-2Dfragment-2Dlist.html&d=DwMFaQ&c=slrrB7dE8n7gBJbeO0g-IQ&r=NUz4Gm9NGJ4L5D1MPwqLQg&m=3diIct9Iwcb1Gtj3QL6m8ifoGHaOaFK28NbCidrbQ2I&s=jFSOANCWv0EHNAcL-AfNi-tDGW-x4oWu_PnZm05TI-U&e=)

<https://boto3.amazonaws.com/v1/documentation/api/latest/reference/services/kinesis-video-archived-media.html#KinesisVideoArchivedMedia.Client.get_media_for_fragment_list>

**KVS and GStreamer Plugin setup - Mac**

1. Please make sure you have all below listed installed and check the version. You can use brew install instead of apt-get install

* Install Git: sudo apt-get install git
* **<code>**$ git --version

git version 2.14.1**</code>**

* Install [CMake](http://kitware.com/cmake" \t "_blank): sudo apt-get install cmake
* **<code>**$ cmake --version

cmake version 3.9.1**</code>**

* Install Libtool: sudo apt-get install libtool

**<code>**2.4.6-2**</code>**

* Install libtool-bin: sudo apt-get install libtool-bin
* **<code>**$ libtool --version
* libtool (GNU libtool) 2.4.6

Written by Gordon Matzigkeit, 1996**</code>**

* Install GNU Automake: sudo apt-get install automake
* **<code>**$ automake --version

automake (GNU automake) 1.15**</code>**

* Install GNU Bison: sudo apt-get install bison
* **<code>**$ bison -V

bison (GNU Bison) 3.0.4**</code>**

* Install G++: sudo apt-get install g++
* **<code>**g++ --version

g++ (Ubuntu 7.2.0-8ubuntu3) 7.2.0**</code>**

* Install curl: sudo apt-get install curl
* **<code>**$ curl --version

curl 7.55.1 (x86\_64-pc-linux-gnu) libcurl/7.55.1 OpenSSL/1.0.2g zlib/1.2.11 libidn2/2.0.2 libpsl/0.18.0 (+libidn2/2.0.2) librtmp/2.3**</code>**

* Install pkg-config: sudo apt-get install pkg-config
* **<code>**$ pkg-config --version

0.29.1**</code>**

* Install Flex: sudo apt-get install flex
* **<code>**$ flex --version

flex 2.6.1**</code>**

2.  Please set the environment variables for library path such as

* + export PATH=/usr/local/Cellar/bison/3.0.4\_1/bin/:$PATH
  + **<code>**export LD\_LIBRARY\_PATH=/opt/awssdk/amazon-kinesis-video-streams-producer-sdk-cpp/kinesis-video-native-build/downloads/local/lib:$LD\_LIBRARY\_PATH**</code>**
  + **<code>**export PATH=**<YourSdkFolderPath>**/kinesis-video-native-build/downloads/local/bin:$PATH**</code>**
  + **<code>**export GST\_PLUGIN\_PATH=**<YourSdkFolderPath>**/kinesis-video-native-build/downloads/local/lib:$GST\_PLUGIN\_PATH**</code>**

Then, I followed the doc and finally it works now. Good Luck.

<https://github.com/awslabs/amazon-kinesis-video-streams-producer-sdk-cpp/blob/master/install-instructions-macos.md#discovering-available-devices>

PS:  Running the gst-launch-1.0 command to start streaming both audio and raw video in Mac-OS

gst-launch-1.0 -v avfvideosrc ! videoconvert ! vtenc\_h264\_hw allow-frame-reordering=FALSE realtime=TRUE max-keyframe-interval=45 ! kvssink name=sink stream-name="my\_stream\_name" access-key="YourAccessKeyId" secret-key="YourSecretAccessKey" osxaudiosrc ! audioconvert ! avenc\_aac ! queue ! sink.

PS2: The default  aws region here is us-west-2. You need to manually set it to different region if you need by adding aws-region="your region" after secret-key.

**KVS Setup Windows (msys2)**

The first step, please download and configure msys2 following the instructions of the document:

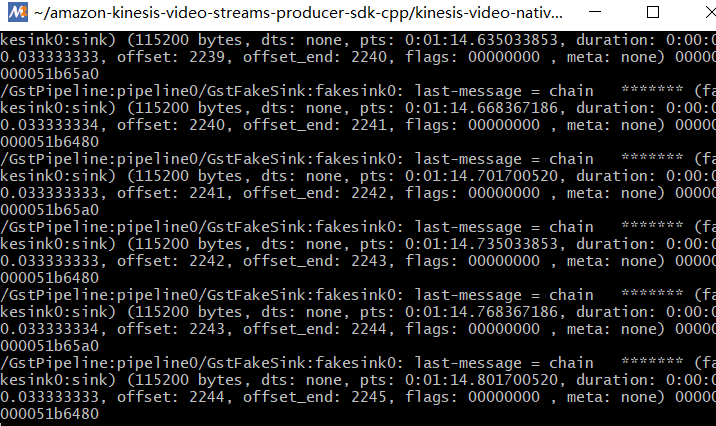
<https://github.com/awslabs/amazon-kinesis-video-streams-producer-sdk-cpp/blob/master/install-instructions-windows-msys2.md>

When you are able to run the command "gst-device-monitor-1.0" to discover the available devices, please stop and don't follow instructions of the document anymore.

The second step, check whether your camera is available by using this command:

**<code>**gst-launch-1.0 -v videotestsrc is-live=TRUE ! fakesink silent=FALSE**</code>**

If the console shows like that



That means your camera is ready for use.

The third step, execute the following command:

gst-launch-1.0 ksvideosrc do-timestamp=TRUE ! video/x-raw,width=640,height=480,framerate=30/1 ! videoconvert ! x264enc bframes=0 key-int-max=45 bitrate=512 ! video/x-h264,profile=baseline,stream-format=avc,alignment=au,width=640,height=480,framerate=30/1 ! kvssink stream-name="YOUR\_STREAM\_NAME" access-key=YOUR\_ACCESS\_KEY secret-key=YOUR\_SECRET\_KEY aws-region="YOUR\_REGION"

If it shows the error like "libgstopencv.dll: could not be found", please use the following command to download or upgrade your dependency.

pacman -U http://repo.msys2.org/mingw/x86\_64/mingw-w64-x86\_64-opencv-4.1.2-2-any.pkg.tar.xz

the version of dependencies can be found at <http://repo.msys2.org/>. If you miss other dependencies, please google to find the corresponding package and do the same. After refreshing all the dependencies, please execute "./min-install-script" to rebuild your plugin.

Attention that the upgrading may fail due to network connection, please keep trying.

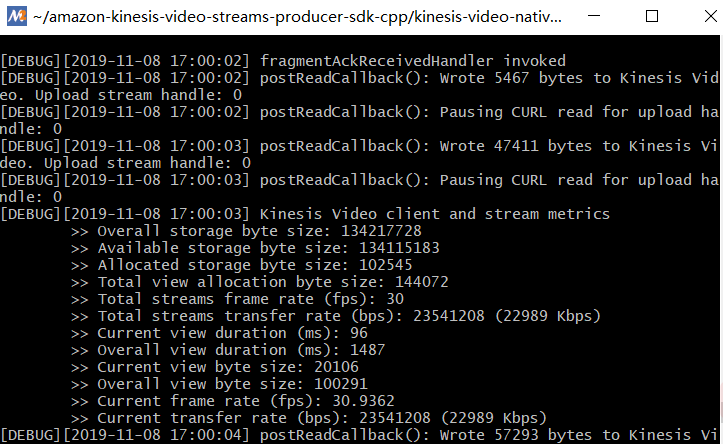
If it shows the error like 'erroneous pipeline: no element "kvssink', please

copy libcproducer.dll and libproducer.dll from C:\msys64\~\amazon-kinesis-video-streams-producer-sdk-cpp\kinesis-video-native-build to C:\msys64\mingw64\bin,

and copy libgstkvssink.dll to C:\msys64\mingw64\lib\gstreamer-1.0.

**Make sure you are not exporting GST\_PLUGIN\_PATH. It is a wrong instruction in the AWS document.**If you have made the export, just close the Mingw console, open it again and go to the ~\amazon-kinesis-video-streams-producer-sdk-cpp\kinesis-video-native-build directory.

Finally, execute the command again, and the console will show like this:



**KVS Setup -Docker**

Following the configuration with docker helped me here.

<https://docs.aws.amazon.com/kinesisvideostreams/latest/dg/examples-gstreamer-plugin.html#examples-gstreamer-plugin-docker-authenticate>

You just have to do 3 steps.

1. Configure Docker with ECR

2. Download kinesis video stream docker image

3. Run docker image of Kinesis video stream

and just launch your camera with a command  inside container

gst-launch-1.0 v4l2src do-timestamp=TRUE device=/dev/video0 ! videoconvert ! video/x-raw,format=I420,width=640,height=480,framerate=30/1 ! x264enc bframes=0 key-int-max=45 bitrate=500 ! video/x-h264,stream-format=avc,alignment=au,profile=baseline ! kvssink stream-name="YourStreamName" storage-size=512 access-key="YourAccessKey" secret-key="YourSecretKey" aws-region="YourAWSRegion"