All the documents and images used to document were done with the software on the Rpi

sudo su

cd /etc

cp hostname hostname.orig
diff hostname hostname.orig
1c1
< pi4-37

> raspberrypi
cp dphys-swapfile dphys-swapfile.orig
diff dphys-swapfile dphys-swapfile.orig
16c16
< CONF_SWAPSIZE=1000

> CONF_SWAPSIZE=100

fetch pkg installers

scp -r pi4-27:~/xx/my-projects-docs/pkg* .

dpkg -l | sort > pkgs.txt

cp pkg-install-scripts/ex* .

adds first set of packages

./extra_pkgs_64bit.sh

dpkg -l | sort > pkgs-a.txt

cp pkg-install-scripts/tensorflow/ex* .

adds 2nd set of packages

./extra-1.sh

dpkg -l | sort > pkgs-b.txt

./extra-2.sh

dpkg -l | sort > pkgs-c.txt

./extra-3.sh

dpkg -l | sort > pkgs-d.txt

./extra-4.sh

dpkg -l | sort > pkgs-e.txt

This is the software to program the picos with SWD

installed-openocd082722-228ede-64bit.img
openocd082722-228ede-64bit.img

Bare Metal for Raspbery Pi

ultibo2.5.123-082722-64bit.img
sudo unsquashfs -d ultibo ultibo2.5.123-082722-64bit.img

qemu-6.2.0-rpios-64bit.img
sudo unsquashfs -d qemu-6.2.0-rpios qemu-6.2.0-rpios-64bit.img
git clone https://github.com/develone/Ultibo_Projects.git
cd Ultibo_Projects/jpeg2000/src/
./compile_ultibo.sh
cd ../QEMU/
./libbuild.sh

vi ~/.local/share/applications/ultibo.desktop
[Desktop Entry]
Name=Lazarus IDE (Ultibo Edition)
Comment=A free pascal platform for bare metal development
Exec=/home/devel/ultibo/core/lazarus.sh
Icon=/home/devel/ultibo/core/images/icons/lazarus.ico
Terminal=false
Type=Application
Categories=Development;IDE;
X-Desktop-File-Install-Version=0.26
scrot -d 3 -s qemujpeg.png
scrot -d 3 -s qemujpeg-1.png
. ~/Ultibo_Projects/picoultibo.sh
/home/devel/ultibo/core:/home/devel/qemu-6.2.0-rpios/bin:/home/devel/local/openocd/bin:/home/devel/picotool/build/:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/local/games:/usr/games

./startqemu.sh

This is what is used to program pico's with SWD.
cd ~/
mkdir local
sudo unsquashfs -d local/openocd/ installed-openocd082722-228ede-64bit.img
Parallel unsquashfs: Using 4 processors
800 inodes (950 blocks) to write

[==|] 950/950 100%

created 800 files
created 33 directories
created 0 symlinks
created 0 devices

```
created 0 fifos
which openocd
/home/devel/local/openocd/bin/openocd
openocd -V
Open On-Chip Debugger 0.11.0-g228ede4-dirty (2022-08-27-19:45)
Licensed under GNU GPL v2
For bug reports, read
        http://openocd.org/doc/doxygen/bugs.html
openocd: invalid option -- 'V'

curl https://pyenv.run | bash
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100   270  100   270    0     0    704      0 --:--:-- --:--:-- --:--:--   703
Cloning into '/home/devel/.pyenv'...
remote: Enumerating objects: 1007, done.
remote: Counting objects: 100% (1007/1007), done.
remote: Compressing objects: 100% (436/436), done.
remote: Total 1007 (delta 581), reused 707 (delta 442), pack-reused 0
Receiving objects: 100% (1007/1007), 495.52 KiB | 3.02 MiB/s, done.
Resolving deltas: 100% (581/581), done.
Cloning into '/home/devel/.pyenv/plugins/pyenv-doctor'...
remote: Enumerating objects: 11, done.
remote: Counting objects: 100% (11/11), done.
remote: Compressing objects: 100% (9/9), done.
remote: Total 11 (delta 1), reused 5 (delta 0), pack-reused 0
Receiving objects: 100% (11/11), 38.72 KiB | 777.00 KiB/s, done.
Resolving deltas: 100% (1/1), done.
Cloning into '/home/devel/.pyenv/plugins/pyenv-installer'...
remote: Enumerating objects: 16, done.
remote: Counting objects: 100% (16/16), done.
remote: Compressing objects: 100% (13/13), done.
remote: Total 16 (delta 1), reused 7 (delta 0), pack-reused 0
Receiving objects: 100% (16/16), 5.88 KiB | 2.94 MiB/s, done.
Resolving deltas: 100% (1/1), done.
Cloning into '/home/devel/.pyenv/plugins/pyenv-update'...
remote: Enumerating objects: 10, done.
remote: Counting objects: 100% (10/10), done.
remote: Compressing objects: 100% (6/6), done.
remote: Total 10 (delta 1), reused 6 (delta 0), pack-reused 0
Receiving objects: 100% (10/10), done.
Resolving deltas: 100% (1/1), done.
Cloning into '/home/devel/.pyenv/plugins/pyenv-virtualenv'...
remote: Enumerating objects: 63, done.
remote: Counting objects: 100% (63/63), done.
remote: Compressing objects: 100% (55/55), done.
remote: Total 63 (delta 11), reused 28 (delta 1), pack-reused 0
Receiving objects: 100% (63/63), 38.44 KiB | 2.75 MiB/s, done.
Resolving deltas: 100% (11/11), done.
Cloning into '/home/devel/.pyenv/plugins/pyenv-which-ext'...
remote: Enumerating objects: 10, done.
remote: Counting objects: 100% (10/10), done.
```

remote: Compressing objects: 100% (6/6), done.
remote: Total 10 (delta 1), reused 6 (delta 0), pack-reused 0
Receiving objects: 100% (10/10), done.
Resolving deltas: 100% (1/1), done.

WARNING: seems you still have not added 'pyenv' to the load path.

Load pyenv automatically by appending
the following to
~/.bash_profile if it exists, otherwise ~/.profile (for login shells)
and ~/.bashrc (for interactive shells) :

export PYENV_ROOT="$HOME/.pyenv"
command -v pyenv >/dev/null || export PATH="$PYENV_ROOT/bin:$PATH"
eval "$(pyenv init -)"

Restart your shell for the changes to take effect.

Load pyenv-virtualenv automatically by adding
the following to ~/.bashrc:

eval "$(pyenv virtualenv-init -)"

These steps save a lot of time installing a lot of python code.
tensorflow
test-1-2.8.img
sudo unsquashfs -d test-1-2.8 test-1-2.8.img
This setup virtual enviornment

cd test-1-28

devel@pi4-37:~/test-1-2.8 $ python3 -m venv env
devel@pi4-37:~/test-1-2.8 $ source env/bin/activate
(env) devel@pi4-37:~/test-1-2.8 $

devel@pi4-37:~/test-1-2.8 $ ipython3 Copy_of_train_hello_world_model.ipynb

```
  0x01, 0x00, 0x00, 0x00, 0x1f, 0x00, 0x00, 0x00, 0x73, 0x65, 0x72, 0x76,
  0x69, 0x6e, 0x67, 0x5f, 0x64, 0x65, 0x66, 0x61, 0x75, 0x6c, 0x74, 0x5f,
  0x64, 0x65, 0x6e, 0x73, 0x65, 0x5f, 0x32, 0x5f, 0x69, 0x6e, 0x70, 0x75,
  0x74, 0x3a, 0x30, 0x00, 0x02, 0x00, 0x00, 0x00, 0x01, 0x00, 0x00, 0x00,
  0x01, 0x00, 0x00, 0x00, 0x0c, 0x00, 0x0c, 0x00, 0x00, 0x00, 0x00, 0x00,
  0x04, 0x00, 0x08, 0x00, 0x0c, 0x00, 0x00, 0x00, 0x14, 0x00, 0x00, 0x00,
  0x04, 0x00, 0x00, 0x00, 0x01, 0x00, 0x00, 0x00, 0x80, 0xff, 0xff, 0xff,
  0xff, 0xff, 0xff, 0xff, 0x01, 0x00, 0x00, 0x00, 0x5d, 0x4f, 0xc9, 0x3c,
  0x04, 0x00, 0x04, 0x00, 0x04, 0x00, 0x00, 0x00
};
unsigned int g_model_len = 2408;
```
(env) devel@pi4-37:~/test-1-2.8 $ exec $SHELL

tensorflow lite
sudo unsquashfs -d project-rpi-tflite project-rpi-tflite102222.img

add to the end of ~/.bashrc

```
export PICO_SDK_PATH=/home/devel/sdk/pico-sdk
export PATH="$HOME/.pyenv/bin:$PATH"
eval "$(pyenv init --path)"
eval "$(pyenv virtualenv-init -)"
```

```
mkdir sdk
```

```
cd sdk
```

This is when the repo is yours.

```
git clone git@github.com:develone/pico-sdk.git
```

With this you can not push changes.

```
git clone https://github.com/develone/pico-sdk.git
```

```
cd pico-sdk/
git submodule update --init
Submodule 'lib/cyw43-driver' (https://github.com/georgerobotics/cyw43-driver.git) registered for path 'lib/cyw43-driver'
Submodule 'lib/lwip' (https://github.com/lwip-tcpip/lwip.git) registered for path 'lib/lwip'
Submodule 'tinyusb' (https://github.com/hathach/tinyusb.git) registered for path 'lib/tinyusb'
Cloning into '/home/devel/sdk/pico-sdk/lib/cyw43-driver'...
Cloning into '/home/devel/sdk/pico-sdk/lib/lwip'...
Cloning into '/home/devel/sdk/pico-sdk/lib/tinyusb'...
Submodule path 'lib/cyw43-driver': checked out '195dfcc10bb6f379e3dea45147590db2203d3c7b'
Submodule path 'lib/lwip': checked out '239918ccc173cb2c2a62f41a40fd893f57faf1d6'
Submodule path 'lib/tinyusb': checked out '4bfab30c02279a0530e1a56f4a7c539f2d35a293'
```

```
cd ../../
```

This is when the repo is yours.

```
git clone git@github.com:develone/devel-pico-tflmicro.git
```

```
git clone https://github.com/develone/devel-pico-tflmicro.git
cd devel-pico-tflmicro
mkdir build
cd build
cmake -DPICO_BOARD=pico .. about 4 hours
Using PICO_SDK_PATH from environment ('/home/devel/sdk/pico-sdk')
PICO_SDK_PATH is /home/devel/sdk/pico-sdk
Defaulting PICO_PLATFORM to rp2040 since not specified.
Defaulting PICO platform compiler to pico_arm_gcc since not specified.
-- Defaulting build type to 'Release' since not specified.
PICO compiler is pico_arm_gcc
-- The C compiler identification is GNU 8.3.1
-- The CXX compiler identification is GNU 8.3.1
-- The ASM compiler identification is GNU
-- Found assembler: /usr/bin/arm-none-eabi-gcc
Build type is Release
PICO target board is pico.
Using board configuration from /home/devel/sdk/pico-sdk/src/boards/include/boards/pico.h
```

-- Found Python3: /usr/bin/python3.9 (found version "3.9.2") found components: Interpreter
TinyUSB available at /home/devel/sdk/pico-sdk/lib/tinyusb/src/portable/raspberrypi/rp2040;
enabling build support for USB.
cyw43-driver available at /home/devel/sdk/pico-sdk/lib/cyw43-driver
lwIP available at /home/devel/sdk/pico-sdk/lib/lwip
-- Configuring done
-- Generating done
-- Build files have been written to: /home/devel/devel-pico-tflmicro/build
make this will take about 4 hours

-rw-r--r-- 1 devel devel 1788264 Oct 25 22:10 libpico-tflmicro.a
-rw-r--r-- 1 devel devel 234456 Oct 25 21:46 libpico-tflmicro_test.a

./pico-sdk/src/rp2_common/boot_stage2/bs2_default.elf
./examples/micro_speech/command_responder_test.elf
./examples/micro_speech/audio_provider_mock_test.elf
./examples/micro_speech/audio_provider_test.elf
./examples/micro_speech/recognize_commands_test.elf
./examples/magic_wand/magic_wand.elf
./examples/magic_wand/gesture_output_handler_test.elf
./examples/magic_wand/magic_wand_test.elf
./examples/magic_wand/gesture_predictor_test.elf
./examples/hello_world/hello_world.elf

This is when the repo is yours.
git clone git@github.com:develone/my-projects-docs.git
git clone https://github.com/develone/my-projects-docs.git

This project uses cmake Important to understand cmake the source
code is 1 level above build.

This is when the repo is yours.
git clone https://github.com/develone/pico-examples -b dev
cd pico-examples
mkdir build

This is when the repo is yours. -b dev is branch dev
git clone --recursive git@github.com:develone/rp2040-freertos-project.git -b dev
git clone --recursive https://github.com/develone/rp2040-freertos-project.git -b dev
cd rp2040-freertos-project/
mkdir build
cd build

cmake -DPICO_BOARD=pico ..
Using PICO_SDK_PATH from environment ('/home/devel/sdk/pico-sdk')
PICO_SDK_PATH is /home/devel/sdk/pico-sdk
Defaulting PICO_PLATFORM to rp2040 since not specified.
Defaulting PICO platform compiler to pico_arm_gcc since not specified.
-- Defaulting build type to 'Release' since not specified.
PICO compiler is pico_arm_gcc
-- The C compiler identification is GNU 8.3.1
-- The CXX compiler identification is GNU 8.3.1

-- The ASM compiler identification is GNU
-- Found assembler: /usr/bin/arm-none-eabi-gcc
Build type is Release
PICO target board is pico.
Using board configuration from /home/devel/sdk/pico-sdk/src/boards/include/boards/pico.h
-- Found Python3: /home/devel/test-1-2.8/env/bin/python3.9 (found version "3.9.2") found components: Interpreter
TinyUSB available at /home/devel/sdk/pico-sdk/lib/tinyusb/src/portable/raspberrypi/rp2040; enabling build support for USB.
cyw43-driver available at /home/devel/sdk/pico-sdk/lib/cyw43-driver
lwIP available at /home/devel/sdk/pico-sdk/lib/lwip
-- Configuring done
-- Generating done
-- Build files have been written to: /home/devel/rp2040-freertos-project/build
make

The elf files a loaded with openocd
. ~/Ultibo/picoultibo.sh
this program a file system performs either klt or dwt lifting step,

openocd -f interface/raspberrypi-swd.cfg -f target/rp2040.cfg -c "program test-read-crc16/test-read-crc16.elf verify reset exit"
ls ../doc lots of documentation
ls ../doc/rp2040-logic-analyzer/rp2040-logic-analyzer.pdf
qpdfview ../doc/rp2040-logic-analyzer/rp2040-logic-analyzer.pdf
./first_pwm/50_pwm.elf
./pico-lifting/pico-lifting.elf
./ultibo_blink/ultibo_blink.elf
./rp2040-logic-analyzer/rp2040-logic-analyzer.elf
./Scheduling/Scheduling.elf
./pico-sdk/src/rp2_common/boot_stage2/bs2_default.elf
./pico-littlefs/e-rw-r--r-- 1 devel devel 1788264 Oct 25 22:10 libpico-tflmicro.a
-rw-r--r-- 1 devel devel 234456 Oct 25 21:46 libpico-tflmicro_test.axample0.elf
./pico-littlefs/example2.elf
./pico-littlefs/example1.elf
./pico-ultibo/pico-ultibo.elf
./test-read/test-read.elf
./ProjectFiles/blink.elf
./kltdwt-ultibo/kltdwt-ultibo.elf
./2tasks/2tasks.elf
./2cores/multicore.elf
./pico-lifting-sf/hello_usb.elf
./test-read-crc16/test-read-crc16.elf
./Mutex/Mutex.elf
./HCSR04/HCSR04.elf
./Semaphore/Semaphore.elf
./klt-test/klt-test.elf

This needed for octave

.octaverc
graphics_toolkit("gnuplot");

https://github.com/develone/svd_rgb.git
cd svd_rgb/src/
devel@pi4-37:~/svd_rgb/src $ make
gcc -c -o obj/svd.o svd.c -I../include
gcc -c -o obj/disp_mat.o disp_mat.c -I../include
gcc -c -o obj/mul_mat.o mul_mat.c -I../include
gcc -c -o obj/pnmio.o pnmio.c -I../include
gcc -c -o obj/error.o error.c -I../include
gcc -c -o obj/mythread.o mythread.c -I../include
gcc -c -o obj/trans_mat.o trans_mat.c -I../include
gcc -c -o obj/master.o master.c -I../include
gcc -o master obj/svd.o obj/disp_mat.o obj/mul_mat.o obj/pnmio.o obj/error.o obj/mythread.o obj/trans_mat.o obj/master.o -I../include -lm -lpthread

./master

octave
In a 2nd shell
scrot -d 3 -s redpgm.png

scrot -d 3 -s rcblu.png

scrot -d 3 -s rcblu-1.png

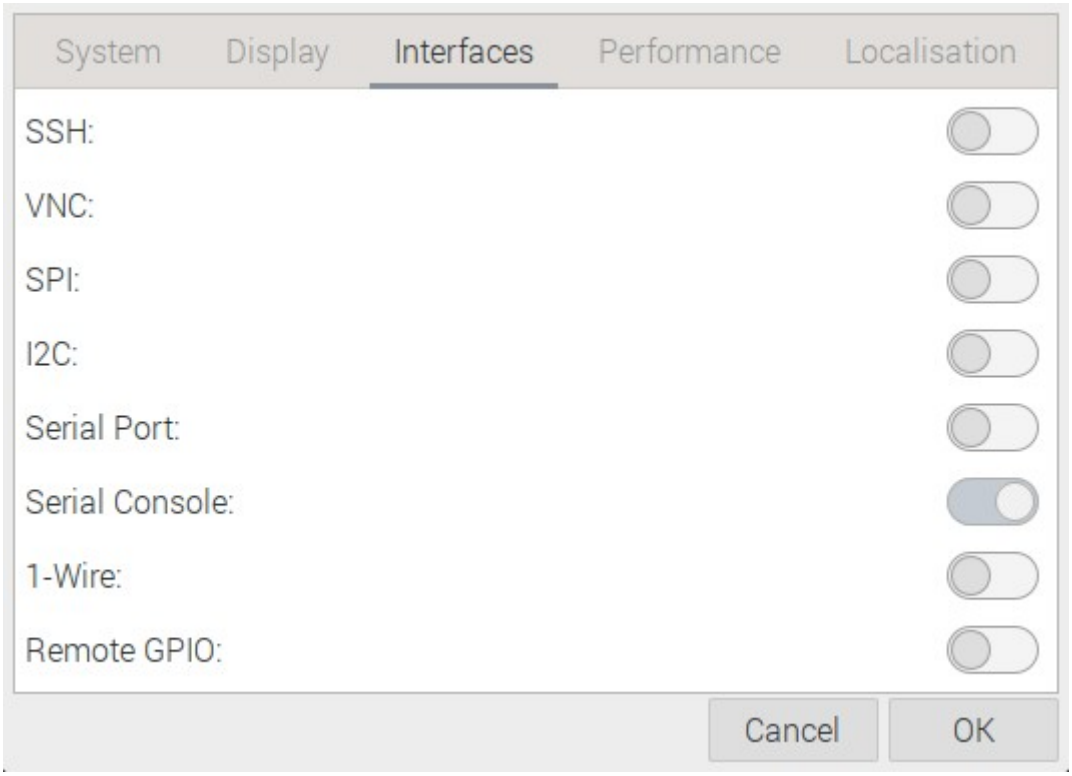
quit

git clone https://github.com/ArduCAM/pico-tflmicro.git
Cloning into 'pico-tflmicro'...
remote: Enumerating objects: 1812, done.
remote: Counting objects: 100% (106/106), done.
remote: Compressing objects: 100% (47/47), done.
remote: Total 1812 (delta 73), reused 59 (delta 59), pack-reused 1706
Receiving objects: 100% (1812/1812), 13.92 MiB | 14.64 MiB/s, done.
Resolving deltas: 100% (950/950), done.

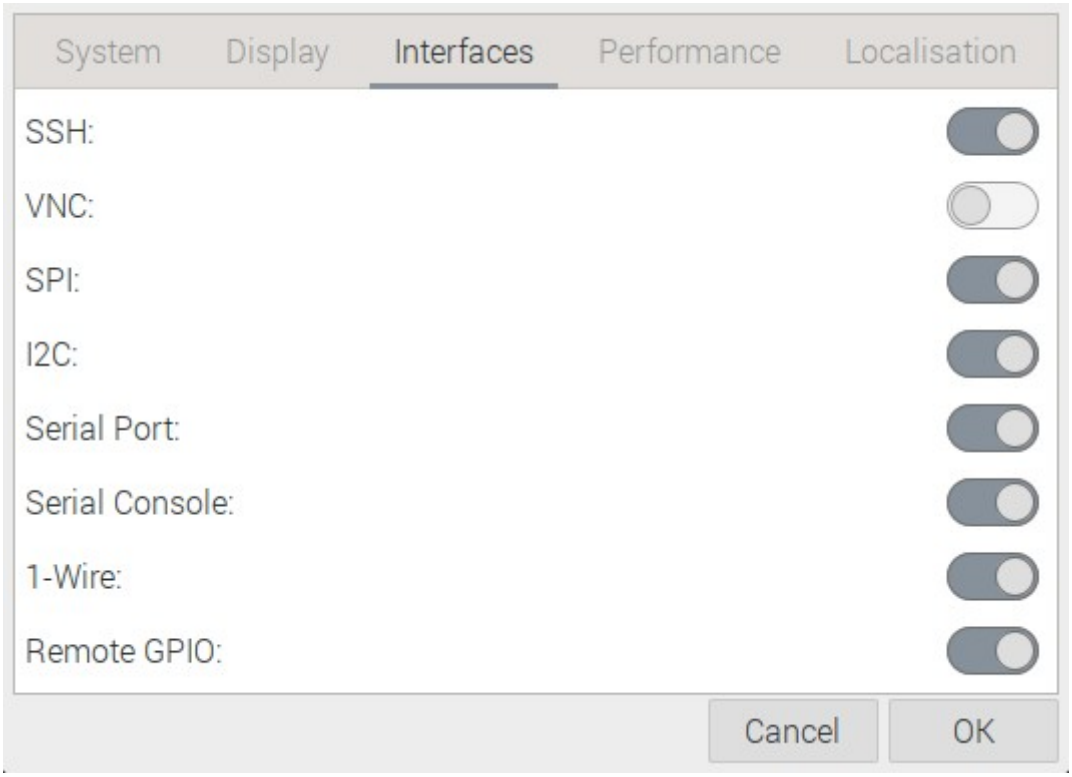
$30.00 at Amazon
devel@pi4-37:~/pico-tflmicro/bin

magic_wand_ble.uf2 person_detection_benchmark.uf2 pico4ml_ble_magic_wand.uf2
micro_speech.uf2   person_detection_int8.uf2   pico4ml_magic_wand.uf2

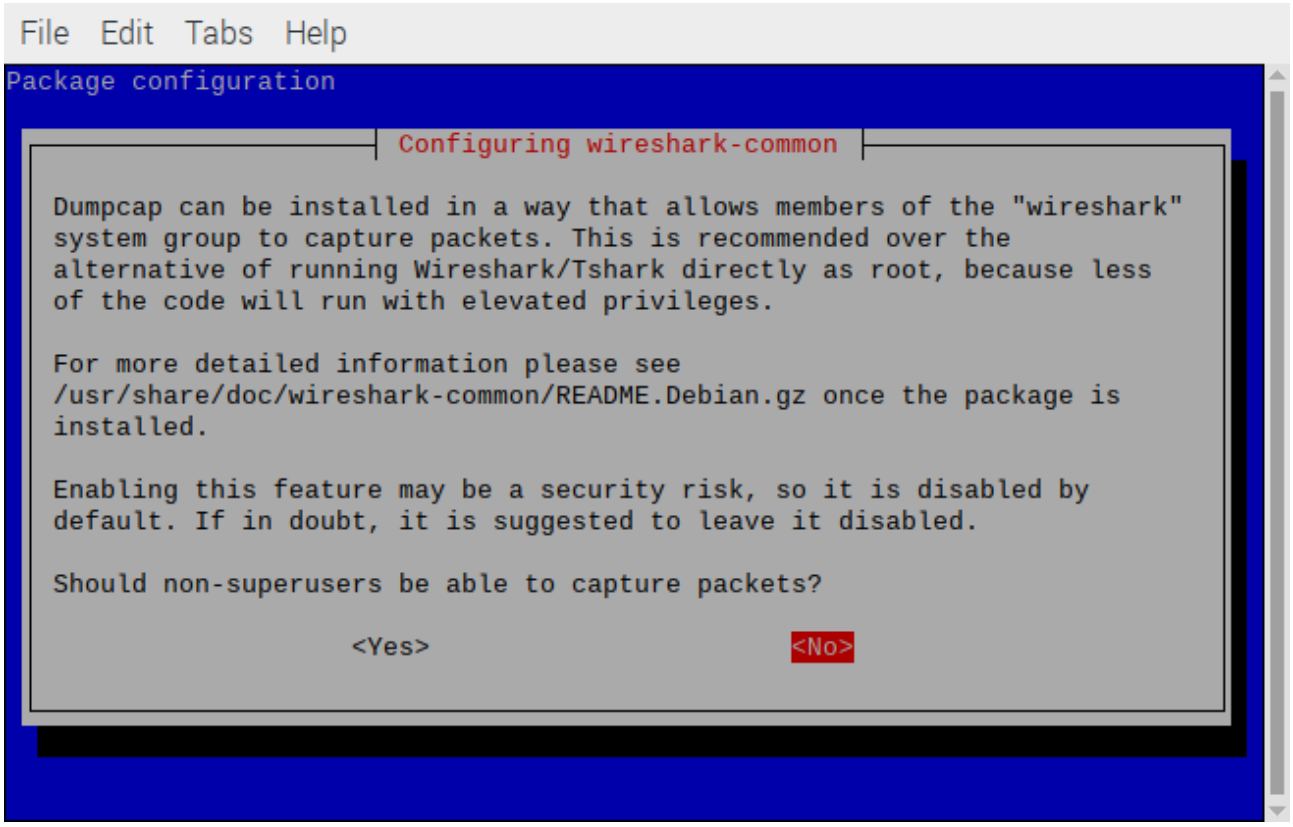
Setting up the interfaces



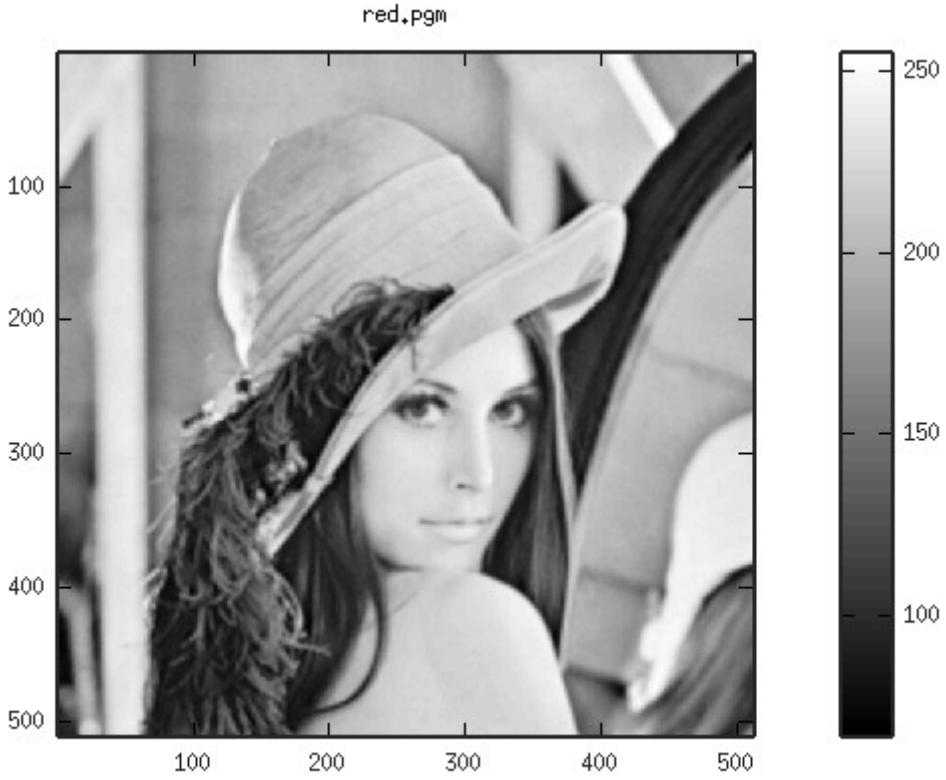
Setting up the interfaces



When installing

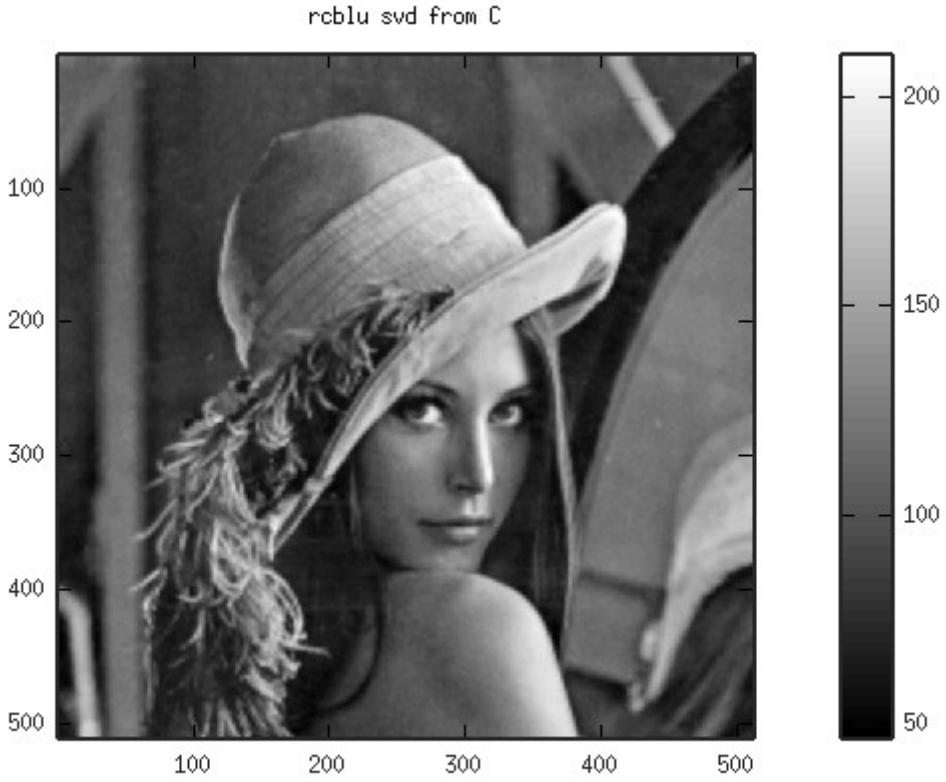


svd_rgb

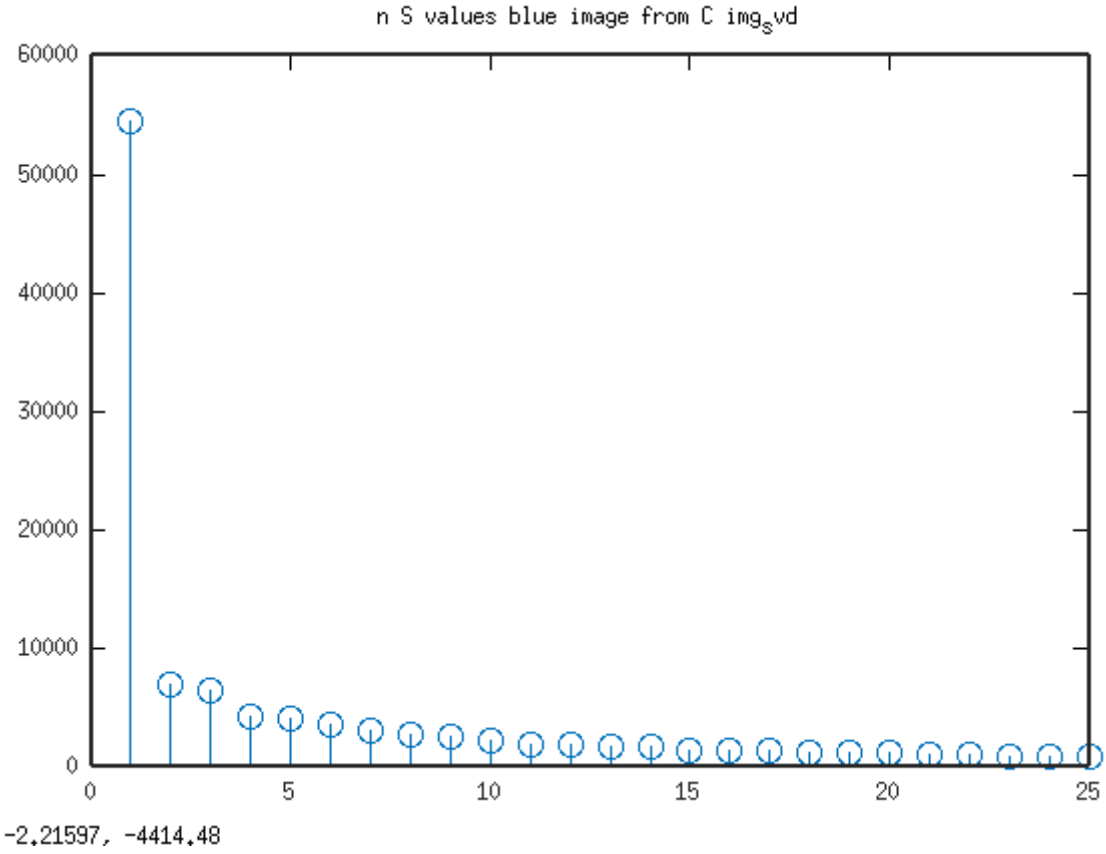


y2= 230.686

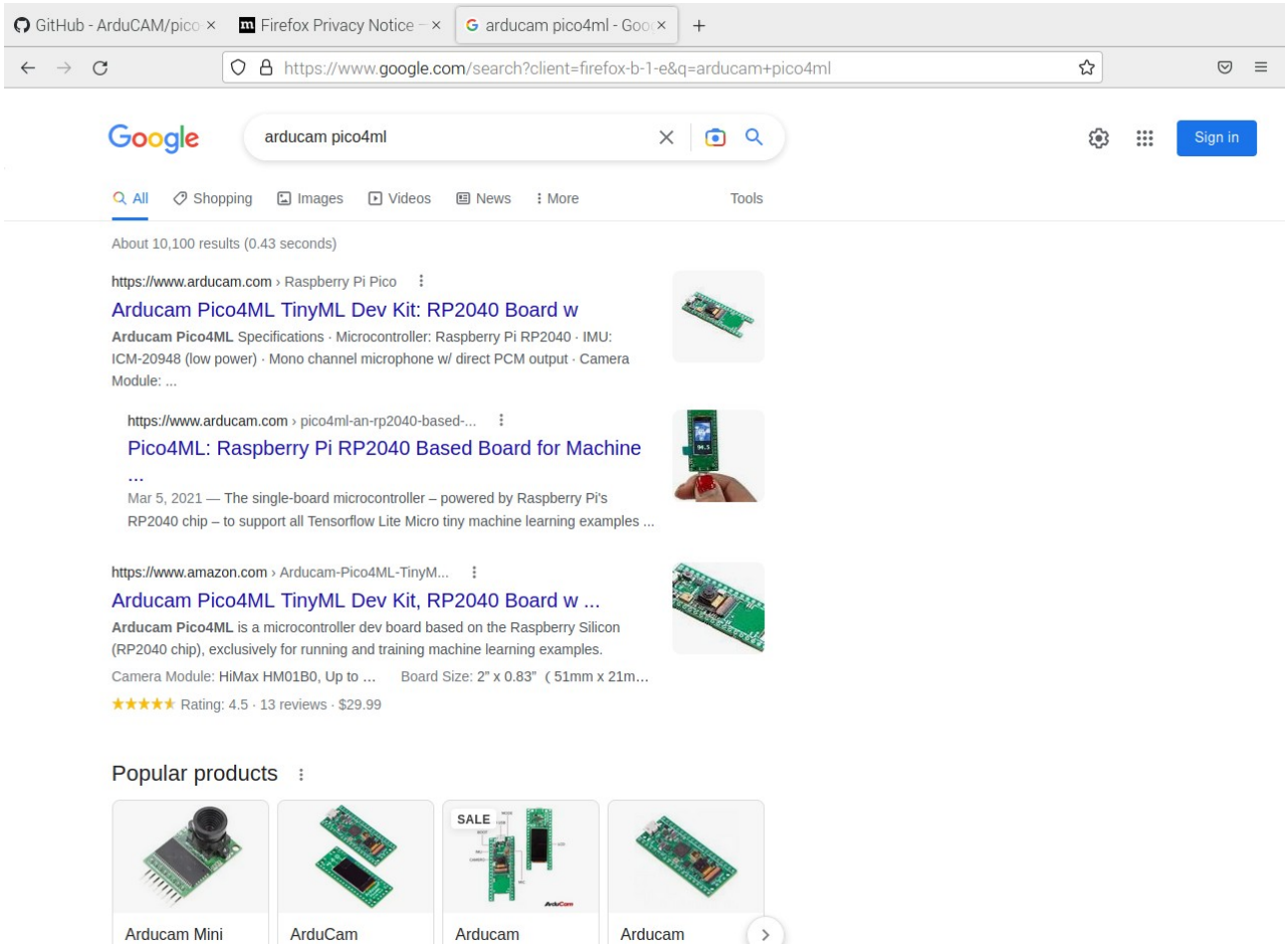
reconstruted



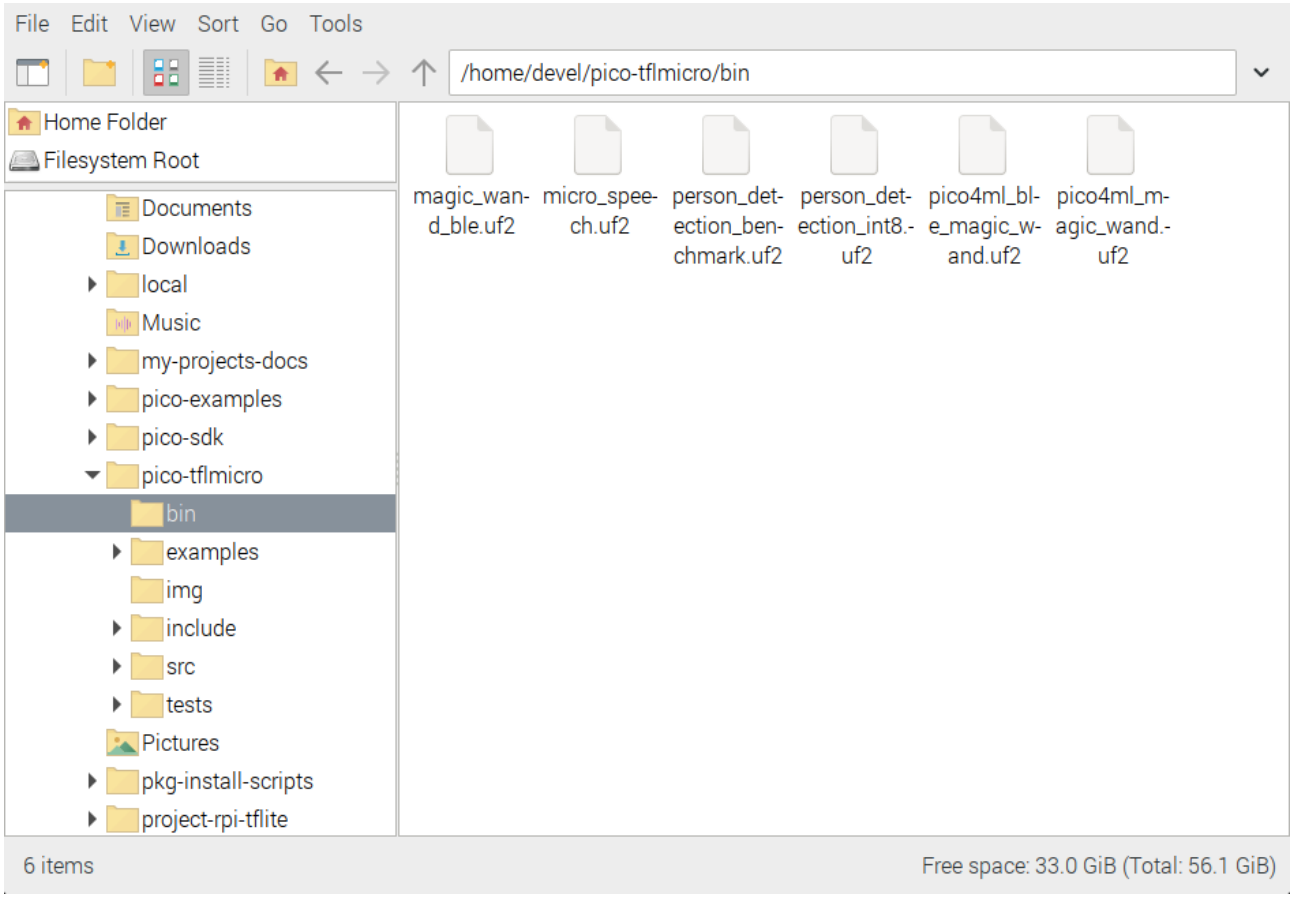
svd



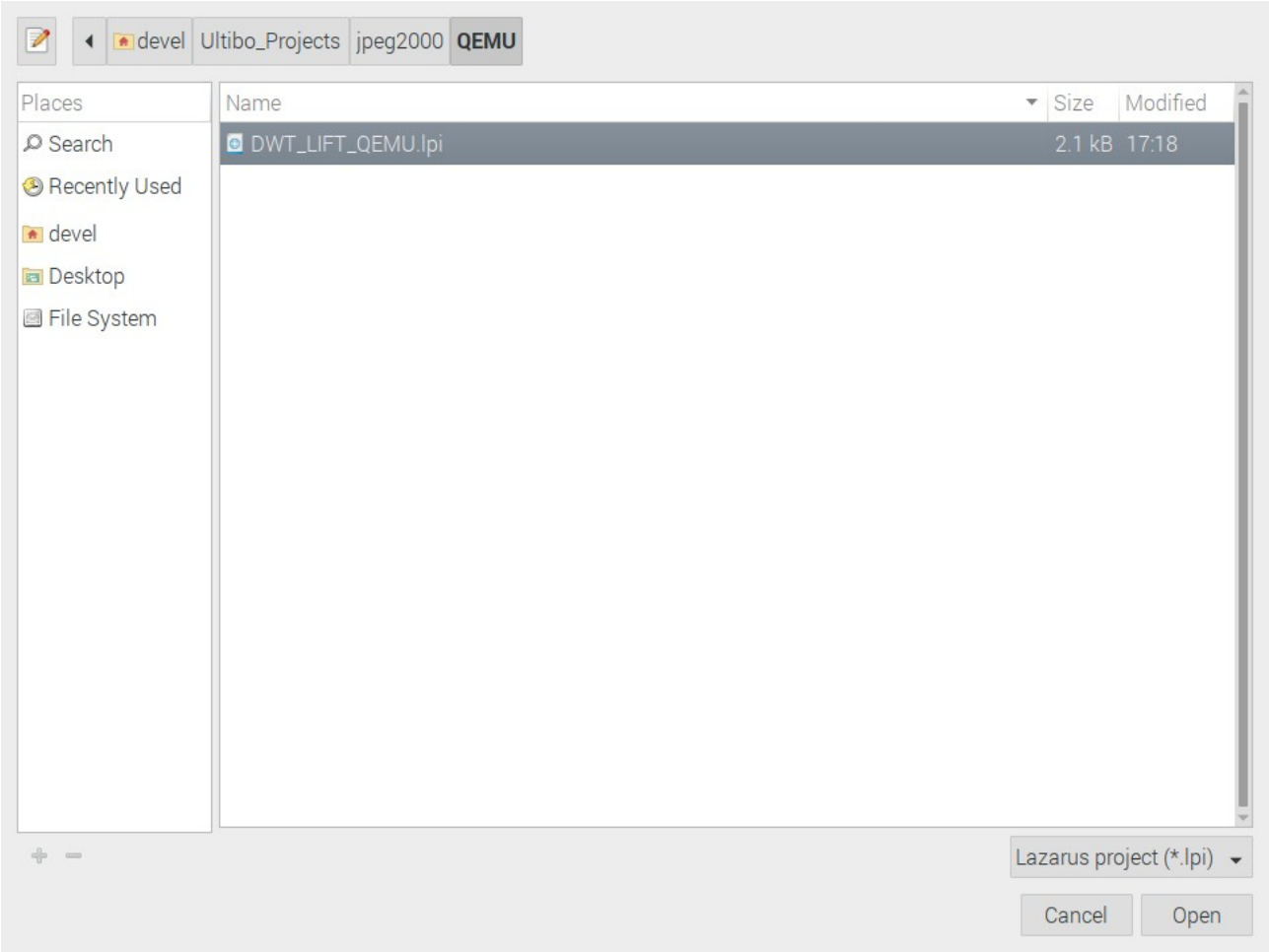
Arducam PicoML


GitHub - ArduCAM/pico
Firefox Privacy Notice
arducam pico4ml - Goo
https://www.google.com/search?client=firefox-b-1-e&q=arducam+pico4ml
Google
arducam pico4ml
Sign in
All
Shopping
Images
Videos
News
More
Tools
About 10,100 results (0.43 seconds)
https://www.arducam.com › Raspberry Pi Pico
Arducam Pico4ML TinyML Dev Kit: RP2040 Board w
Arducam Pico4ML Specifications · Microcontroller: Raspberry Pi RP2040 · IMU: ICM-20948 (low power) · Mono channel microphone w/ direct PCM output · Camera Module: ...
https://www.arducam.com › pico4ml-an-rp2040-based-...
Pico4ML: Raspberry Pi RP2040 Based Board for Machine ...
Mar 5, 2021 — The single-board microcontroller – powered by Raspberry Pi's RP2040 chip – to support all Tensorflow Lite Micro tiny machine learning examples ...
https://www.amazon.com › Arducam-Pico4ML-TinyM...
Arducam Pico4ML TinyML Dev Kit, RP2040 Board w ...
Arducam Pico4ML is a microcontroller dev board based on the Raspberry Silicon (RP2040 chip), exclusively for running and training machine learning examples.
Camera Module: HiMax HM01B0, Up to ...
Board Size: 2" x 0.83" (51mm x 21m...
Rating: 4.5 · 13 reviews · $29.99
Popular products
SALE
Arducam Mini
ArduCam
Arducam
Arducam

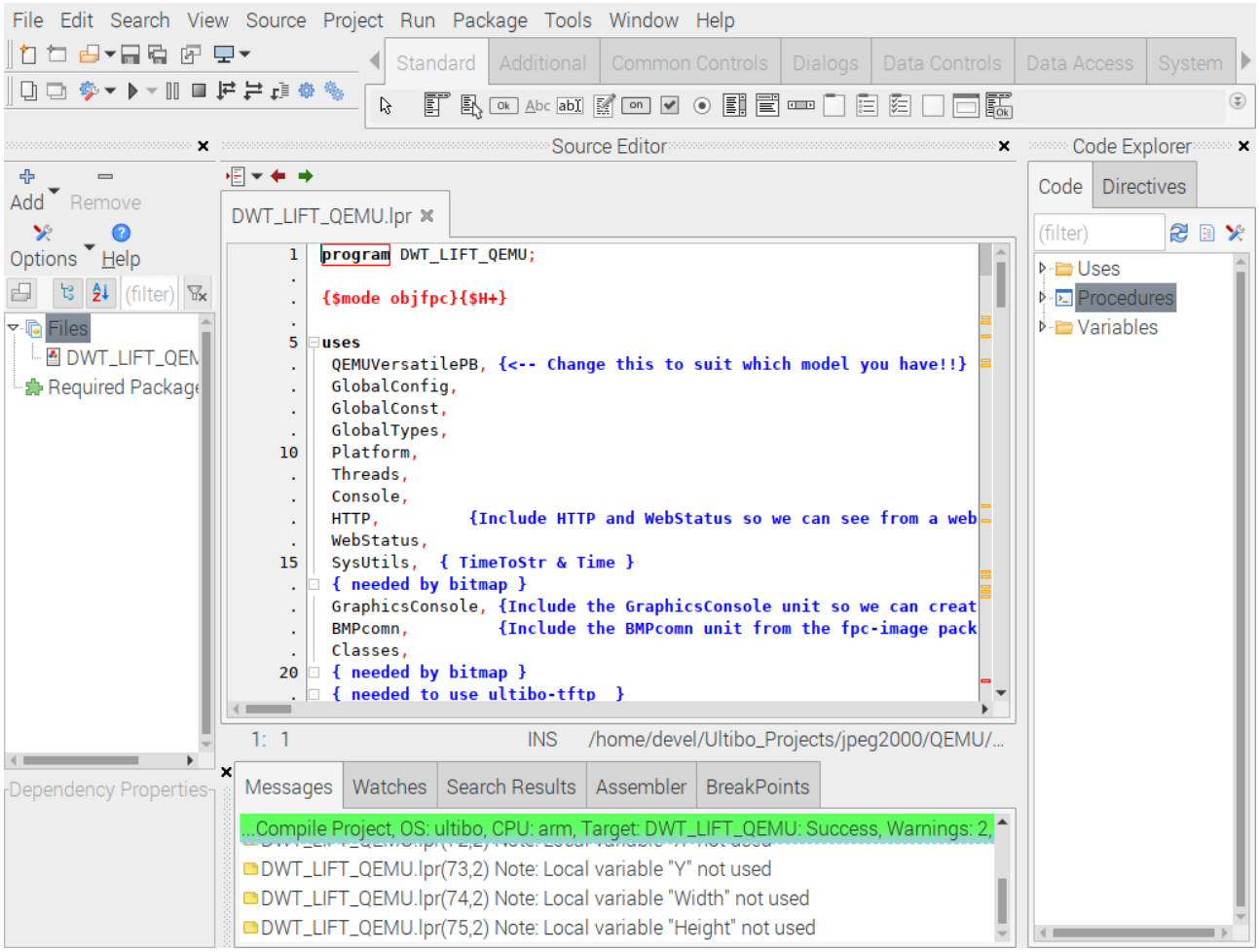
UF2


File Edit View Sort Go Tools
/home/devel/pico-tflmicro/bin
Home Folder
Filesystem Root
Documents
Downloads
local
Music
my-projects-docs
pico-examples
pico-sdk
pico-tflmicro
bin
examples
img
include
src
tests
Pictures
pkg-install-scripts
project-rpi-tflite
magic_wand_ble.uf2
micro_speech.uf2
person_detection_benchmark.uf2
person_detection_int8.uf2
pico4ml_ble_magic_wand.uf2
pico4ml_magic_wand.uf2
6 items
Free space: 33.0 GiB (Total: 56.1 GiB)

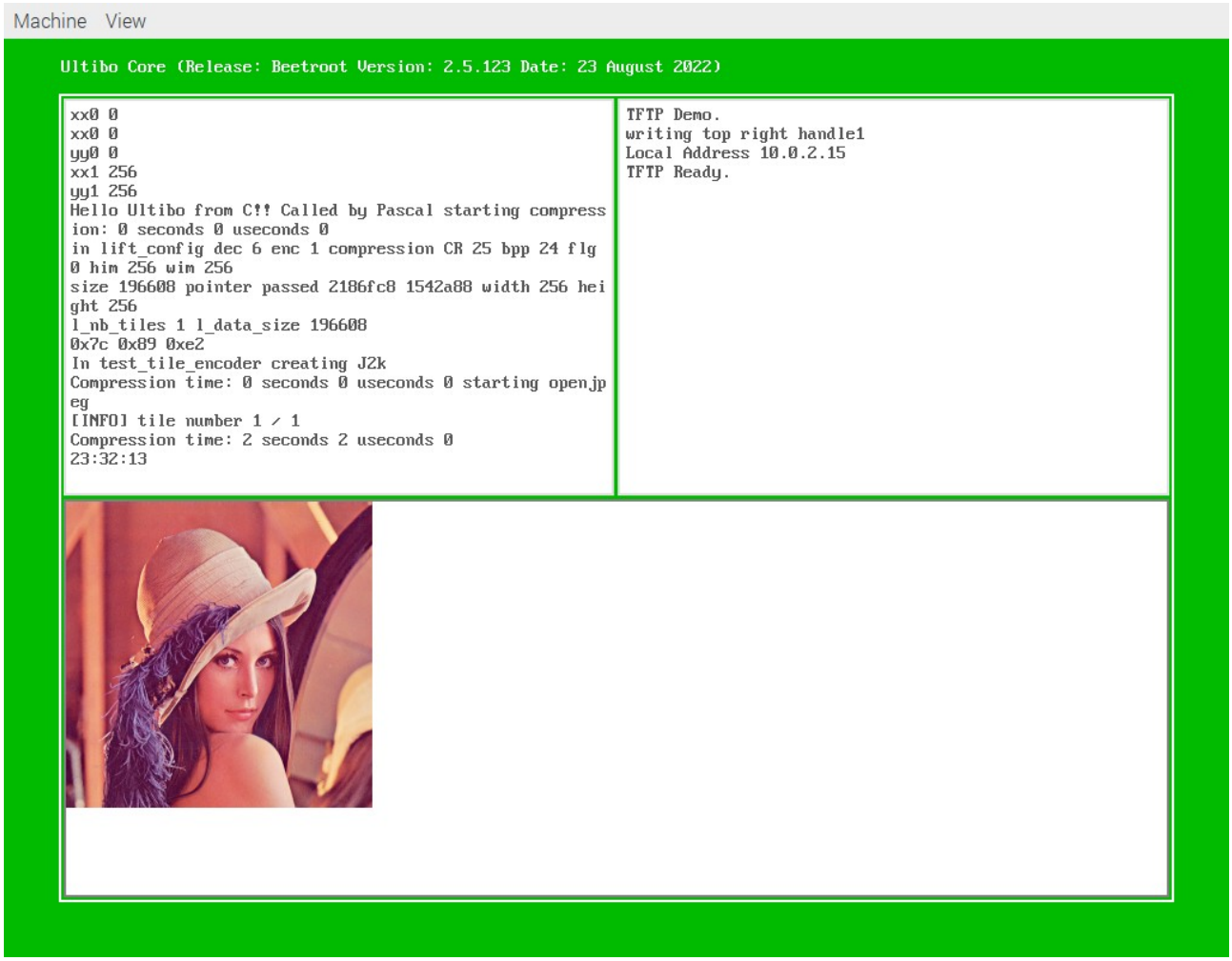
QEMU


devel
Ultibo_Projects
jpeg2000
QEMU
Places
Search
Recently Used
devel
Desktop
File System
Name
Size
Modified
DWT_LIFT_QEMU.lpi
2.1 kB
17:18
Lazarus project (*.lpi)
Cancel
Open

QEMU


File Edit Search View Source Project Run Package Tools Window Help
Standard Additional Common Controls Dialogs Data Controls Data Access System
Source Editor
Code Explorer
Add Remove
Options Help
(filter)
Files
DWT_LIFT_QEM
Required Package
Dependency Properties
DWT_LIFT_QEMU.lpr
1 program DWT_LIFT_QEMU;
{$mode objfpc}{$H+}
uses
QEMUVersatilePB, {<-- Change this to suit which model you have!!}
GlobalConfig,
GlobalConst,
GlobalTypes,
Platform,
Threads,
Console,
HTTP, {Include HTTP and WebStatus so we can see from a web
WebStatus,
SysUtils, { TimeToStr & Time }
{ needed by bitmap }
GraphicsConsole, {Include the GraphicsConsole unit so we can creat
BMPcomn, {Include the BMPcomn unit from the fpc-image pack
Classes,
{ needed by bitmap }
{ needed to use ultibo-tftp }
1: 1 INS /home/devel/Ultibo_Projects/jpeg2000/QEMU/...
Messages Watches Search Results Assembler BreakPoints
...Compile Project, OS: ultibo, CPU: arm, Target: DWT_LIFT_QEMU: Success, Warnings: 2,
DWT_LIFT_QEMU.lpr(73,2) Note: Local variable "Y" not used
DWT_LIFT_QEMU.lpr(74,2) Note: Local variable "Width" not used
DWT_LIFT_QEMU.lpr(75,2) Note: Local variable "Height" not used
Code Directives
(filter)
Uses
Procedures
Variables

QEMU



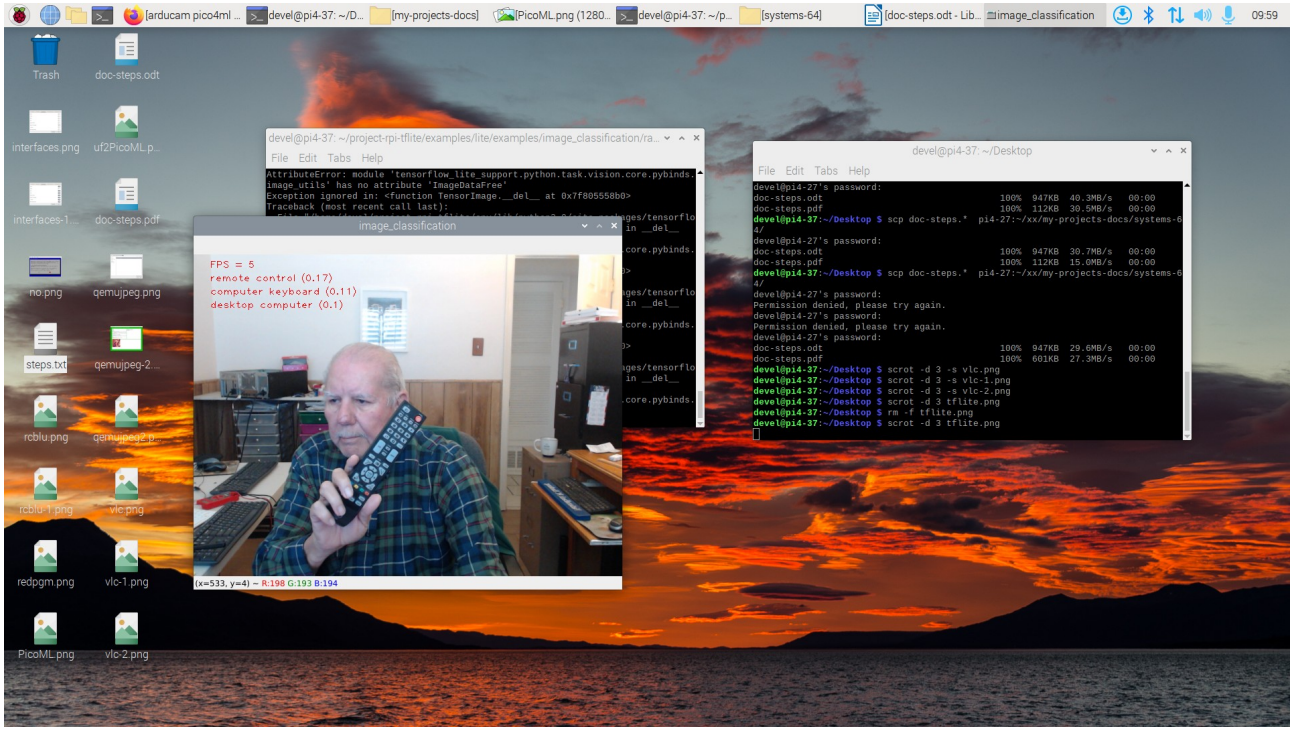
Starting here will be in an update.

cd ~/project-rpi-tflite/
python3 -m venv env
source env/bin/activate
(env) devel@pi4-37:~/project-rpi-tflite $

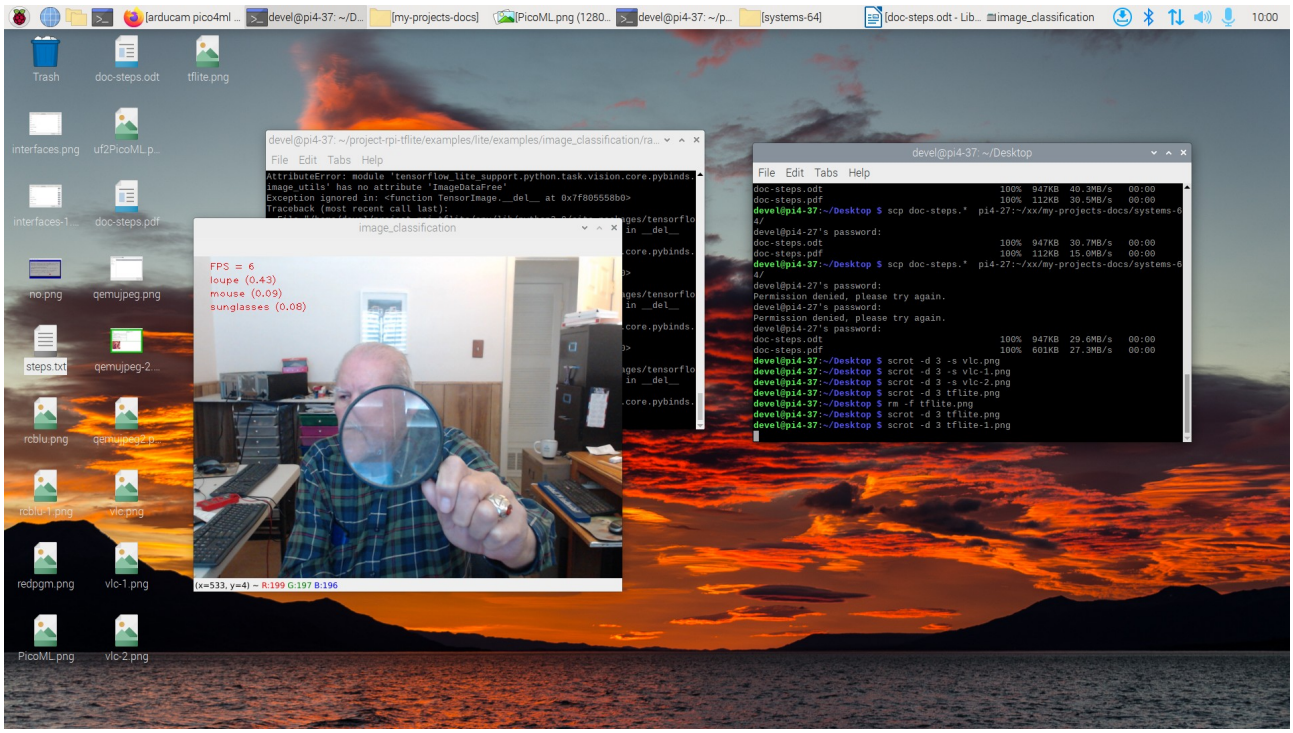
cd examples/lite/examples/image_classification/raspberry_pi/

python3 classify.py

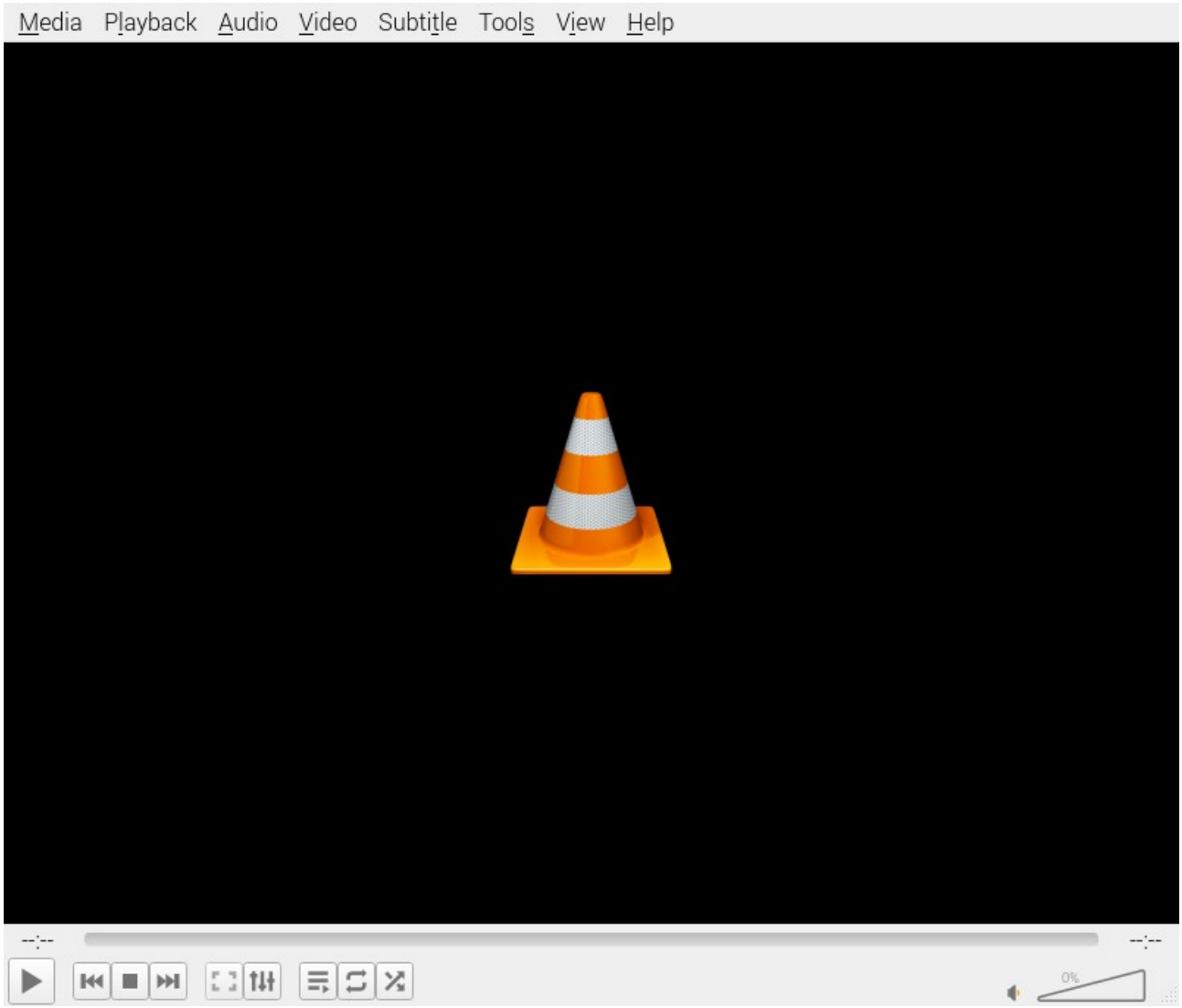
TensorFlow Lite detects remote control



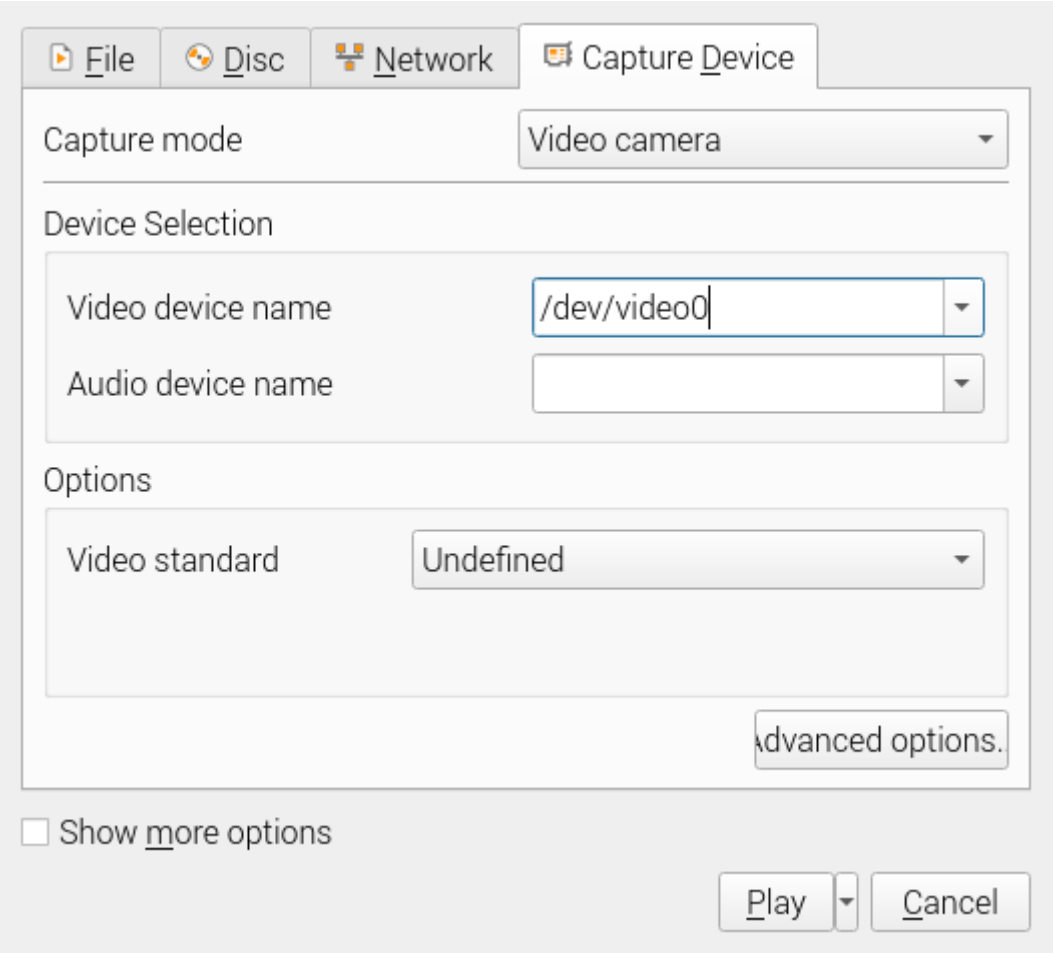
TensorfFlow Lite detects loupe



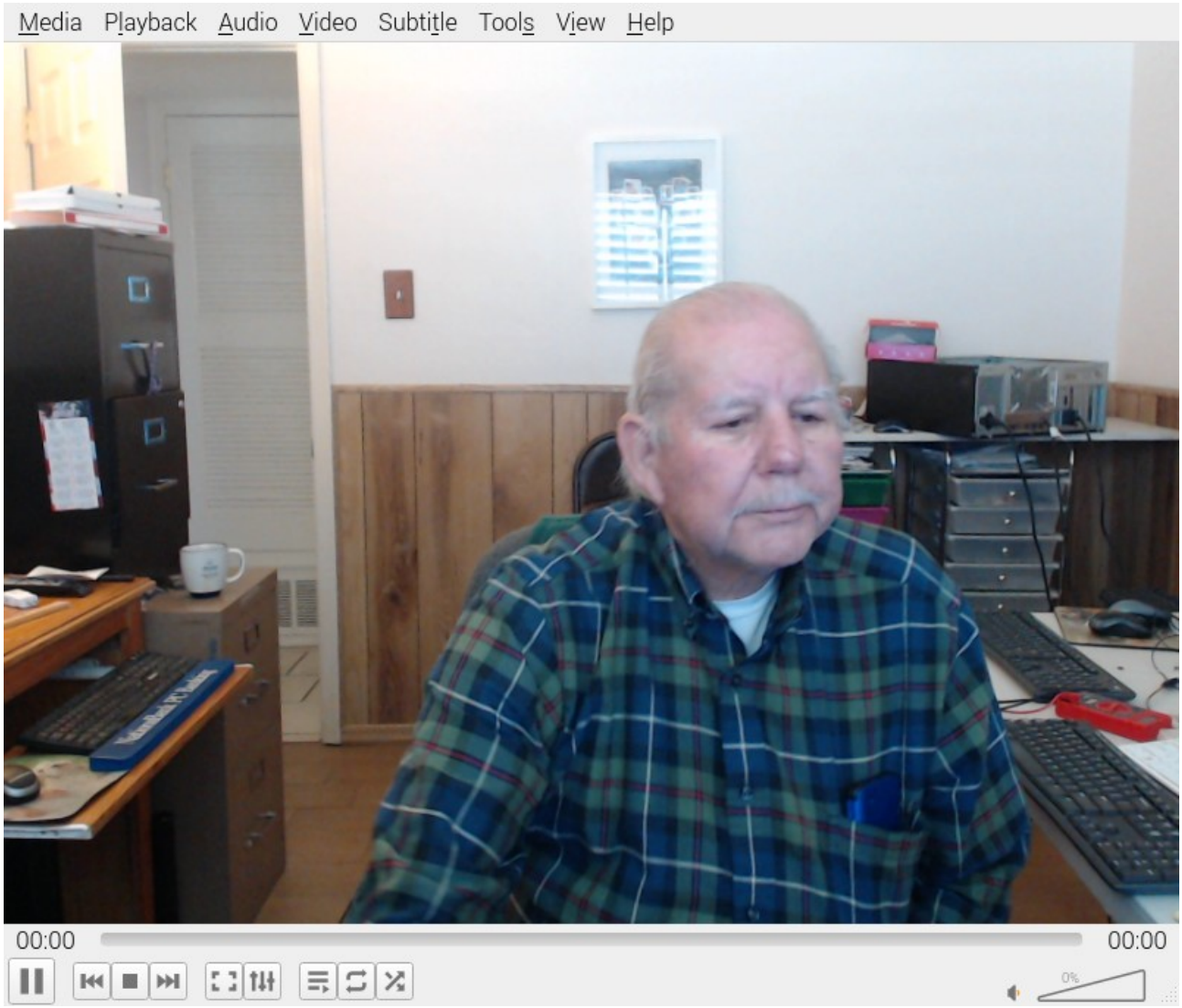
vlc & camera



selecting the video device


File
Disc
Network
Capture Device
Capture mode
Video camera
Device Selection
Video device name
/dev/video0
Audio device name
Options
Video standard
Undefined
dvanced options.
Show more options
Play
Cancel

me in my lab


Media Playback Audio Video Subtitle Tools View Help
00:00
00:00
0%

xx