Speed Enforcement – Application Notes

Overview

This application note describes how to develop a velocity control device based on the 24 GHz uRAD solutions.

The idea consists of monitoring with great accuracy the velocity of vehicles that drive on an urban or interurban road. The velocity of vehicles approaching and moving away from the radar can be monitored.

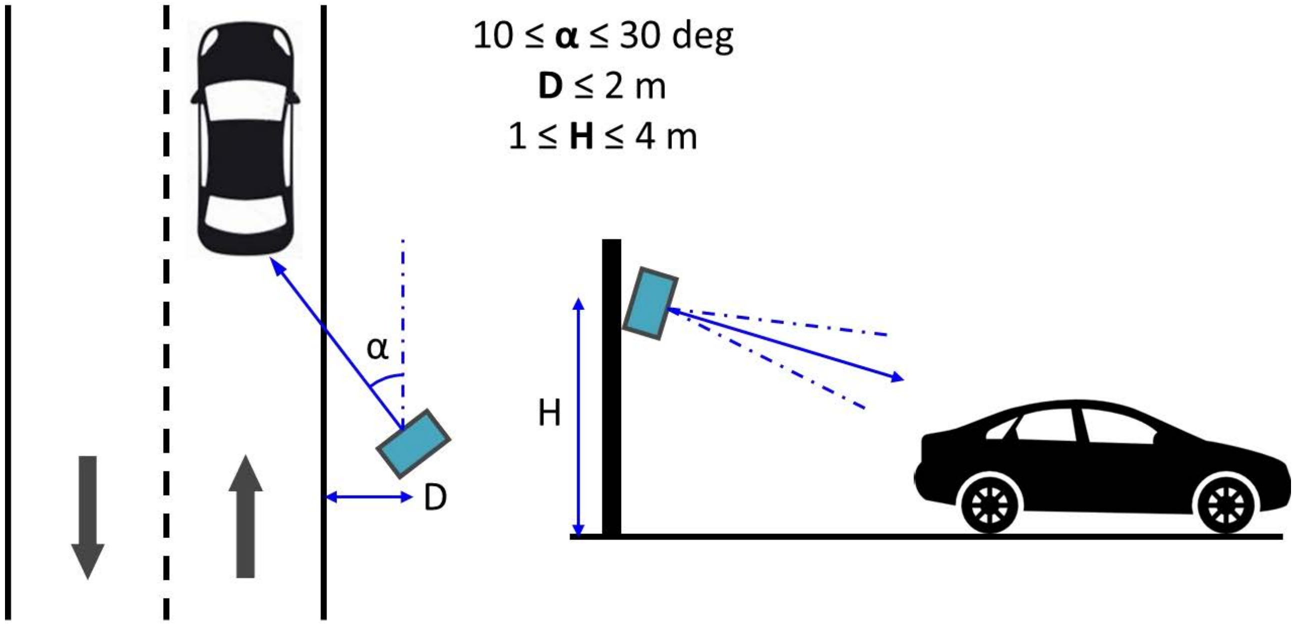
Devices

This application can be developed with any of 24 GHz uRAD solutions: uRAD Arduino, uRAD Raspberry Pi or uRAD USB. We chose these solutions for two main reasons in their performance:

- Ability to operate in continuous wave mode, allowing to measure velocities up to 270 km/h with high accuracy.
- Narrow field of view equal to 30 x 30 degrees.

Mounting

The suggested setup consists of placing the radar on the side of the road following these recommendations:



- At a distance **D** from the edge of the road less than 2 meters. It is preferable to detect vehicles from behind, since, in general, the shape of the rear of the vehicles favors the reflection of the radar wave.

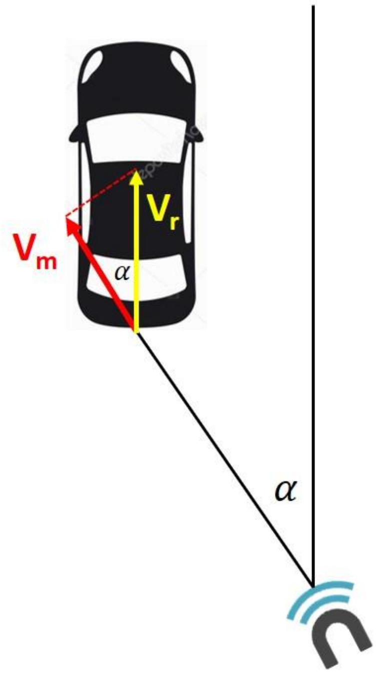
- With a positioning angle α close to 10 and with a maximum of 30 degrees. The greater the angle, the greater the imprecision measured in the velocity, as described in the next point.
- At a height **H** between 1 and 4 meters. If it is placed at a height much higher than the height of the vehicle, slightly tilt the radar downwards.

Theoretical aspects

The main theoretical aspect to take into account is that the measured velocity is always the radial velocity, that is, the component of velocity that falls on the line that joins the radar and the vehicle.

In the following image, the velocity measured by uRAD is $\mathbf{V_m}$, however, the real velocity of the vehicle is $\mathbf{V_r}$. The real velocity is obtained by dividing by the cosine of the angle.

$$\text{Velocity}_{real} = \text{Velocity}_{measured} / \cos\alpha$$



If the positioning angle is low, the error made is small. For example, for a car driving at 100 km/h, with α = 10 degrees, uRAD measures 98.48 km/h. For a 30 degrees angle, uRAD measures 86.60 km/h, which results in a larger error.

Therefore, when a car passes in front of the radar, its velocity will be measured at different angles α. When the vehicle is closer to uRAD, the angle will be greater and the measured velocity more inaccurate, and as it moves away, the angle will be smaller and the measured velocity more accurate. We will see this clearly a little later in the example with real data.

Configuration

The suggested initial configuration is the same as in the *uRAD_velocity_meter* example, and it is valid for any of the 24 GHz uRAD models. In this case the radar software/firmware must be SDK 1.1.

- **mode = 1**: continuous wave Doppler mode. The best mode to measure only velocity.
- **f0 = 125**: emits at 24.125 GHz, in the center of the band.
- **BW = 240**: irrelevant parameter in mode 1 since it only emits at one frequency.
- **Ns = 200**: 200 samples to have the best accuracy.
- **Ntar = 3**: detect the velocity of up to three targets.
- **Vmax = 75**: a value of 75 corresponds to search along the entire range of available velocities (75 m/s = 270 km/h).
- **MTI = 0**: irrelevant parameter in mode 1.
- **Mth = 0**: irrelevant since we do not want boolean movement information.
- **Alpha = 20**: we define the initial Alpha sensitivity at 20.
- **distance_true = False**: distance information cannot be received in mode 1.
- **velocity_true = True**: receive velocity information.
- **SNR_true = True**: receive information of Signal-to-Noise Ratio.
- **I_true = False**: we are not interested in RAW data of the in-phase component of the IF signal.
- **Q_true = False**: we are not interested in RAW data of the quadrature component of the IF signal.
- **movement_true = False**: we are not interested in boolean indicator of whether movement has been detected or not.

This initial configuration is only a recommendation, as the parameters must be adjusted according to your particular scenario. All configuration parameters are described in detail in the user manual. The most relevant parameters to be adjusted are:

- **Alpha** adjusts the sensitivity. The minimum is 3 and the maximum is 25. A lower value increases the sensitivity, but also the ghost detections.
- **Ntar** selects the number of targets to search. Velocity results are ordered from highest to lowest SNR detected. Starting with up to 3 targets is a good option to see real detections and also ghost detections if any. Also, if you want to detect vehicles in both directions at the same time.

Program an example

The following code is also based on the example script *uRAD_velocity_meter* provided with the purchase. In this case, in addition to displaying the results on the screen, what we do is to save the velocity and SNR results obtained along with their timestamp in a text file, and then draw them on a graph over time.

The code corresponds to the script programmed in Python, but an alternative example is also available for Arduino. Only the most relevant part of code is displayed.

```
# switch ON uRAD
return_code = uRAD_USB_SDK11.turnON(ser)
if (return_code != 0):
    closeProgram()

if (not usb_communication):
    sleep(timeSleep)

# loadConfiguration uRAD
return_code = uRAD_USB_SDK11.loadConfiguration(ser, mode, f0, BW, Ns, Ntar, Vmax, M
TI, Mth, Alpha, distance_true, velocity_true, SNR_true, I_true, Q_true, movement_tr
ue)
if (return_code != 0):
    closeProgram()

if (not usb_communication):
    sleep(timeSleep)

resultsFileName = 'velocity.txt'
fileResults = open(resultsFileName, 'a')

# infinite detection loop
while True:

    # target detection request
    return_code, results, raw_results = uRAD_USB_SDK11.detection(ser)
    if (return_code != 0):
        closeProgram()

    # Extract results from outputs
    NtarDetected = results[0]
    velocity = results[2]
    SNR = results[3]

    t_i = time()

    velocity_string = ''
    # Iterate through desired targets
```

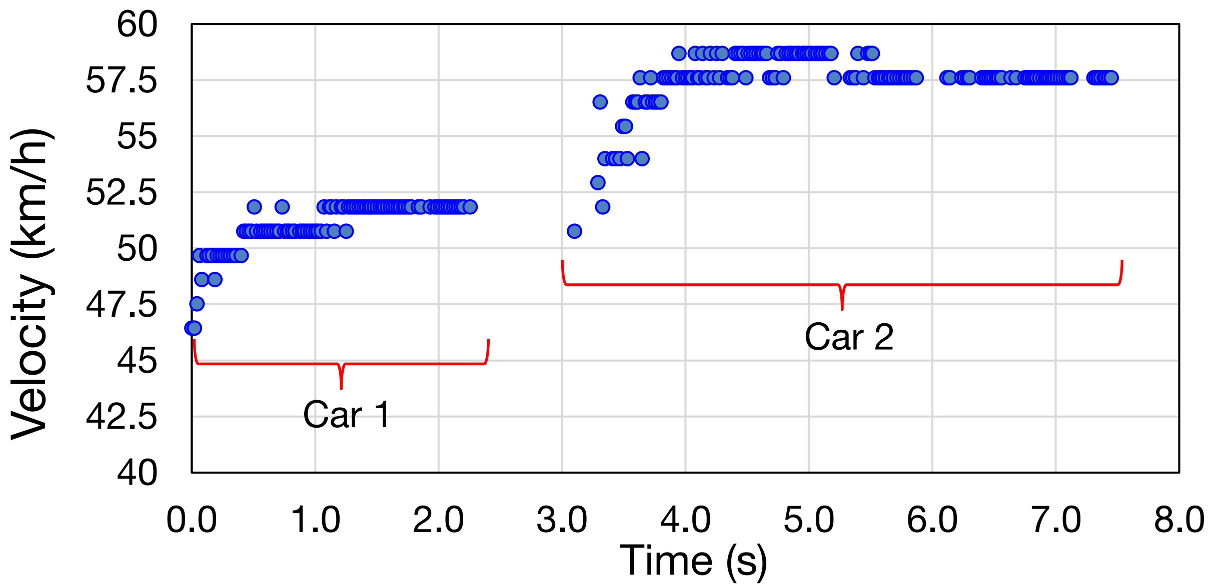
```
    for i in range(NtarDetected):
        # If SNR is big enough
        if (SNR[i] > 0):
            # Prints target information
            print("Target: %d, Velocity: %1.1f m/s, SNR: %1.1f dB" % (i+1, velocity[i], SNR[i]))

            # Save target information
            velocity_string += ('%1.1f %1.1f ' % (velocity[i], SNR[i]))

    if (NtarDetected > 0):
        fileResults.write(velocity_string + '%1.3f\n' % t_i)
        print(" ")
```

Real measurements

Representative results of actual measurements are shown below.

The radar is positioned as in the following image, at a height of about 90 cm, at one meter from the road and at an angle of approximately 10 degrees.

The following graph shows the velocity results captured by the radar of two cars that have passed with a difference of a little less than a second between them.



It is observed in both cars how the first samples are measured with a greater error due to the relative angle between the radar and the vehicle, and as the vehicle moves away, the velocity stabilizes until its real value, 51.84 km/h for the first car and 57.6 km/h for the second.

The accuracy in the measurement is ±1.08 km/h (±0.3 m/s).

In this case, the speed is positive, indicating that the cars are moving away from the radar. Cars moving away from (positive velocities) and approaching (negative velocities) the radar can be monitored at the same time, which is why it would be valid for a two-lane scenario with different direction of driving.