

Forum:[Hack Forums](#)

Thread: [RTL-SDR passive motion sensing \(FM illuminator Doppler\) not X-ray vision](#)

itcorp - July 29, 2026, 1:25 pm

I keep seeing tik tok clips like "cops/military can look through walls and tell how many people are in a room.".I got curious,dug into what that tech actually is,and built a small cheap demo so it stops sounding like movie magic.

So i made small hackforums rtl tracking demo...

First: what those videos are usually about

They're not showing a normal camera that "sees through concrete."

The real idea is closer to radar/RF sensing:

- send (or use) radio waves
- stuff in the room reflects them
- if a person moves, the reflection changes (Doppler/multipath)

Real systems are expensive:UWB,mmWave,multi-antenna Wi-Fi CSI,proper radar boxes,lots of calibration.

TikTok often mixes that with thermal cams, audio, or straight CGI.

So yeah...the principle is real.The "phone app X-ray vision" vibe usually isn't.

Quick note on government/LE devices

Publicly discussed "through-wall" style gear (military/police/rescue) is usually in a completely different class from a USB TV stick:

They transmit on purpose (active radar/UWB/mmWave),or run multi-antenna/CSI setups

They often estimate range + motion,sometimes with a screen that looks more like blobs or range-Doppler than a camera feed

Built for specific jobs:hostage rooms,clear/hold,disaster search with training,calibration,and legal limits

Prices and licensing are nothing like hobby SDR.You're talking dedicated product lines,not a weekend WPF app

So if tiktok shows a rugged box telling you "2 people moving behind that wall," that's closer to purpose-built through-wall radar/UWB sensing than "software on an RTL-SDR."

Same family of physics (RF reflection + Doppler).Completely different hardware,DSP budget,and claims.

My demo is closer to a classroom experiment of the idea not a substitute for those boxes.

What I built and what it is NOT

I did not build a through-wall scanner.

I built a cheap receiver demo:

Tune the RTL-SDR to a strong FM radio station (like a lamp lighting the room).

When you walk around, reflections change a little.

The app looks for that tiny "wiggle" in the walking frequency range (~0.5–12 Hz).

If the wiggle is strong enough => Person motion.

Important limits:

this does not draw a person behind a wall

"people count" is a rough guess (0–3), not real tracking

if FM is weak in your house, detection will suck

Think of it like this:

FM station = light in the room

your body moving = light reflections change

RTL-SDR = cheap sensor watching those changes

Doppler panel = where motion shows up

RF panel = only checks if the "light" (FM peak) is strong enough

Code from the project (basic detection idea)

Same pipeline as above, but from the actual C# files.

1) IQ bytes => complex samples "IqConverter.cs"

Code //RTLTracking/Dsp/IqConverter.cs

```
public static Complex[] ToComplex(byte[] raw)
```

```
{
```

```
    int n = raw.Length / 2;
```

```
    var samples = new Complex[n];
```

```
    double sumI = 0, sumQ = 0;
```

```
    for (int i = 0; i < n; i++)
```

```
    {
```

```
        double ii = (raw[2 * i] - 127.5) / 127.5;
```

```
        double qq = (raw[2 * i + 1] - 127.5) / 127.5;
```

```
        samples[i] = new Complex(ii, qq);
```

```
        sumI += ii;
```

```
        sumQ += qq;
```

```
    }
```

```
    var mean = new Complex(sumI / n, sumQ / n);
```

```
    for (int i = 0; i < n; i++)
```

```
        samples[i] -= mean;
```

```
    return samples;
```

```
}
```

2) Envelope + downsample to ~125 Hz "DopplerMotionDetector.cs"

Walking is slow. Sample rate is huge. So take "iq", average blocks, push into a ring buffer.

Code //RTLTracking/Dsp/DopplerMotionDetector.cs -> Process()

```

_envAccum += Math.Sqrt(re * re + im * im);
_envCount++;

if (_envCount < _decimFactor) continue;

double sample = _envAccum / _envCount;
_envAccum = 0;
_envCount = 0;

_ring[_ringWrite] = sample;
_ringWrite = (_ringWrite + 1) % DopplerFftSize;
if (_ringFill < DopplerFftSize) _ringFill++;

if (!produced || _ringFill < DopplerFftSize)
    return new DopplerMotionResult(0, false, 0, 0, 0, false, null);

return Analyze();
3) Doppler FFT + walk band vs noise "Analyze()"

```

FFT the envelope window. Energy in 0.5–12 Hz vs higher band. Ratio over threshold => motion.

```

Code //RTLTracking/Dsp/DopplerMotionDetector.cs -> Analyze()
Fourier.Forward(buf, FourierOptions.Matlab);

```

```

// band = mean power in 0.5–12 Hz
// noise = mean power around 15–40 Hz
_noiseEma = 0.92 * _noiseEma + 0.08 * noise;
_bandEma = 0.70 * _bandEma + 0.30 * band;

```

```

double ratio = _bandEma / Math.Max(_noiseEma, 1e-18);
bool rawMotion = ratio >= _threshold;

```

```

int peaks = CountPeaks(power, loBin, hiBin, _noiseEma * Math.Max(3.0, _threshold));
int movers = Math.Clamp(_moversEma, 0, 3); // rough estimate, not a census
4) UI loop "MainViewModel.cs"

```

```

Code //RTLTracking/ViewModels/MainViewModel.cs -> ProcessLoop()
var iq = IqConverter.ToComplex(raw);
latest = _detector.Process(iq);

```

```

if (iq.Length >= _fft.FftSize)
    PushWaterfallRow(_fft.ComputeMagnitudeDb(iq)); // RF: FM peak check

```

```

if (dopplerRow != null)
    PushDopplerRow(dopplerRow, motion); // Doppler: 0–20 Hz motion
RF waterfall = "is the lamp (FM) on?"
Doppler waterfall = "is someone moving?"

```

How the detection idea works (simple)
 RTL-SDR gives I/Q samples (not audio).
 Convert bytes -> complex numbers ("IqConverter").
 Take signal strength over time (envelope = "I + Iq").
 Human walking is slow, so we downsample that envelope to ~125 Hz.

Run an FFT on that slow signal (Doppler FFT).
Compare energy in 0.5–12 Hz (walking) vs higher bands (noise-ish).
If walking energy is high enough -> motion.

Tiny code idea for step 1:

```
Code //bytes come as I0,Q0,I1,Q1,... (127.5 ~= zero)
double i = (raw[2 * n] - 127.5) / 127.5;
double q = (raw[2 * n + 1] - 127.5) / 127.5;
var sample = new Complex(i, q);
Tiny code idea for "is someone moving?"
```

```
Code // band = energy in 0.5–12 Hz
// noise = energy higher up
double ratio = band / noise;
bool motion = ratio >= threshold;
"Number of people" is just counting peaks in that walk band and clamping to 0–3. Useful as a vibe
check. Not a headcount.
```

XtremeDiscount - July 29, 2026, 1:39 pm

Wow, pretty interesting how you can track movement with FM. Cool post!

itcorp - July 29, 2026, 1:40 pm

(Jul 29, 2026 01:39 PM)XtremeDiscount Wrote: [\[CITE\]](#) Wow, pretty interesting how you can track movement with FM. Cool post!

Thanks man. Yeah, FM is basically just the "lamp" your body walking around messes with the multipath a bit, and the Doppler side picks up that slow wiggle. It's nowhere near real through-wall LE gear, but for a cheap RTL-SDR experiment it's surprisingly fun once you land on a strong station. Glad you liked the post! I really appreciate your opinion!

with more rtl devices even better things can be done