

# Pigeon Sentry v5

X-10 TURBOLASER EDITION · KAMERA-ZIELERKENNUNG · GEZIELTES FEUER · INTERNER WASSERTANK

Projekt-Dossier · DIY-Hardware · Home-Assistant-Integration · Standort Fürth 90762 · Ziel: Stadttaube (*Columba livia domestica*, *Fraktion: Rebellenabschaum*)

■ WAS SICH GEGENÜBER v4 ÄNDERT

Drei große Umbauten: **(1) Kamera-Zielerkennung** nach dem Taubenturret-Prinzip — Pi Zero 2 W mit Kameramodul, Bewegungserkennung on-device, KI-Bestätigung durch ein externes Backend, Berechnung der Pan/Tilt-Winkel aus den 2D-Pixelkoordinaten. Statt v4-Zufallszonen wird jetzt **gezielt** geschossen (Reaktionszeit ~500 ms bei gutem WLAN). **(2) Der Actor-ESP32 aus v4 entfällt** — Servos und Pumpenrelais wandern an den Pi, der ohnehin zielen muss. Der Audio-ESP32-S3 (Edge Impulse) bleibt als zweiter Erkennungskanal und Prä-Trigger erhalten. **(3) Das komplette System zieht in den X-10 Turbolaser-Tower** (Dice-Tower-STL, auf 200 % skaliert, ASA-Druck): Wassertank + Pumpe + Elektronik in der Basis, Kamera + Düse im drehbaren Geschützkopf. Kein Außenhahn, kein Magnetventil, kein Gartenschlauch quer über die Terrasse — der Turm ist autark.

|                          |                          |                                    |                          |                    |
|--------------------------|--------------------------|------------------------------------|--------------------------|--------------------|
| MATERIALKOSTEN<br>~230 € | BAUZEIT<br>3 Wochenenden | DRUCKZEIT<br>~60–80 h (200 %, ASA) | REAKTIONSZEIT<br>~500 ms | HA-ENTITIES<br>12+ |
|--------------------------|--------------------------|------------------------------------|--------------------------|--------------------|

Bauplan / Inhalt

| #  | Abschnitt                                                | Aufwand        | Status   |
|----|----------------------------------------------------------|----------------|----------|
| 01 | Systemarchitektur: Audio + Video + gezieltes Feuer       | 10 min Lektüre | NEU      |
| 02 | Stückliste v5 (Delta zu v4)                              | ~230 € gesamt  | GEÄNDERT |
| 03 | X-10 Tower: STL-Analyse, Skalierung, Umbau, Druck        | 60–80 h Druck  | NEU      |
| 04 | Wassertank, Pumpe & Wasserweg                            | 60 min         | NEU      |
| 05 | Elektrik & Verkabelung (Pi statt Actor-ESP32)            | 90 min         | GEÄNDERT |
| 06 | Software: Taubenturret auf dem Pi + Backend auf dem MeLE | 90 min         | NEU      |
| 07 | Audio-ESP32: unverändert, neue Rolle als Prä-Trigger     | 15 min         | GEÄNDERT |
| 08 | Home Assistant: Entities, Kamera-Stream, Automationen    | 45 min         | GEÄNDERT |
| 09 | Kalibrierung: Zielen, Parallaxe, Offsets                 | 45 min         | GEÄNDERT |
| 10 | Betrieb: Winter, Sonne, Regen, Realismus                 | laufend        | GEÄNDERT |

01 Systemarchitektur NEU IN v5

v5 kombiniert die bewährte Audio-Erkennung aus v3/v4 mit dem Kamera-Ansatz des Taubenturret-Projekts. Es gibt jetzt **drei Rechenknoten** mit klar getrennten Rollen:

| Knoten               | Hardware                                                 | Aufgabe                                                                                      | Firmware / Software                             |
|----------------------|----------------------------------------------------------|----------------------------------------------------------------------------------------------|-------------------------------------------------|
| Audio-Node           | ESP32-S3 + INMP441                                       | Hört Gurren, Edge-Impulse-Inferenz, publiziert MQTT (unverändert aus v4)                     | Arduino-Sketch v4, OTA                          |
| Vision- & Actor-Node | Pi Zero 2 W + Kameramodul v3 + 2 Servos + Relais + Pumpe | Bewegungserkennung, Backend-Abfrage, Winkelberechnung aus Pixelkoordinaten, zielt und feuert | Taubenturret (github.com/ MLWeber/taubenturret) |
| Erkennungs-Backend   | MeLE Quieter 4C (dein Debian-/Docker-Host)               | KI-Objekterkennung: bestätigt „Vogel“ auf dem Snapshot, liefert Bounding Box                 | taubenturret-backend als Docker-Container       |

**Warum der Actor-ESP32 rausfliegt:** In v4 hat der ESPHome-Chip stumpf Zufallszonen angefahren. Sobald aber eine Kamera die Taube *lokalisiert*, muss derselbe Rechner, der die Pixelkoordinaten kennt, auch die Servos steuern — sonst schleppst du Koordinaten über drei Systeme mit entsprechender Latenz. Der Taubenturret-Stack macht genau das bereits: Kamera → Bewegung → Backend-Bestätigung → 2D-Pixel → 3D-Winkel → Servo → Relais. Der Pi übernimmt also Vision *und* Aktorik. Weniger Chips, weniger MQTT-Hops, ~500 ms Gesamtlatenz.

Datenfluss

```
KANAL A · VIDEO (primär, zielgenau)
  Kamera (Pi, picamera2) -> Bewegungserkennung on-device
  -> Snapshot an Backend (MeLE, Docker) via WLAN
  -> Backend: "Vogel, Konfidenz 0.94, BBox [x,y,w,h]"
  -> Pi rechnet Pixel -> Pan/Tilt-Winkel (inkl. Offsets)
  -> Servos schwenken -> Relais -> Pumpe -> 600 ms Wasserstoss
  -> MQTT: pigeon/shot +1, Videoclip wird gespeichert

KANAL B · AUDIO (sekundär, Prä-Trigger & Redundanz)
  Audio-ESP32 hört Gurren, p=0.93 -> MQTT pigeon/detected
  -> HA-Automation -> REST-Call an Pi: "Alarmmodus"
  -> Pi erhöht Framerate/Sensitivität für 60 s
  -> Taube läuft ins Bild -> Kanal A übernimmt
```

SENSOR-FUSION = WENIGER FALSE POSITIVES

v4 hat bei jedem verdächtigen Geräusch blind in Zufallszonen geschossen — Marder, Wind, tiefe Stimmen inklusive. v5 feuert nur noch, wenn die **Kamera das Ziel bestätigt und lokalisiert** hat. Das Audio-Signal dient als Frühwarnung (Taubе gurrт oft, bevor sie im Sichtfeld sitzt) und als Redundanz für den Sichtschatten. Erwartung: False-Positive-Rate sinkt um eine Größenordnung, Wasserverbrauch ebenso — wichtig, weil der Tank jetzt endlich ist.

VORAUSSETZUNGEN

MQTT-Broker in HA (Mosquitto, wie v4) · MeLE Quieter 4C mit Docker und freien ~2 GB RAM für das Backend · stabiles 2,4-GHz-WLAN am Aufstellort, besser 5 GHz per USB-Adapter am Pi (der Zero 2 W kann nativ nur 2,4 GHz und ist der Latenz-Flaschenhals Nr. 1).

## 02 Stückliste v5 (Delta zu v4)

Aus v4 übernommen werden Audio-ESP32-S3, Mikrofon, Netzteile, Relais und Kleinmaterial. **Entfallen:** Actor-ESP32, Magnetventil, Gardena-Kupplungen, ½"-Schlauch. **Neu:** Pi, Kamera, Pumpe, Tank, Lager, Filament.

### Bleibt aus v4 (bereits vorhanden)

| St. | Bauteil                                         | Rolle in v5                                                                    | ≈ Preis |
|-----|-------------------------------------------------|--------------------------------------------------------------------------------|---------|
| 1×  | ESP32-S3 DevKit (PSRAM)                         | Audio-Node, unverändert                                                        | 15 €    |
| 1×  | INMP441 I2S-Mikrofon + Windschutz               | unverändert                                                                    | 9 €     |
| 1×  | 1-Kanal-Relaismodul 5 V, Optokoppler, aktiv-LOW | schaltet jetzt die Pumpe statt des Ventils                                     | 4 €     |
| 2×  | MG996R Servos                                   | Pan/Tilt — reichen; optional 15-kg-Servos (B09KZ8VTNB) für den schwereren Kopf | 14 €    |
| 1×  | Netzteil 12 V/2 A + Netzteil 5 V/3 A            | 12 V → Pumpe · 5 V → Pi, Servos, ESP32                                         | 22 €    |
| 1×  | Elko 1000 µF, Hohlstecker-Adapter, Jumper       | unverändert                                                                    | 10 €    |

### Neu in v5

| St.  | Bauteil                                                                    | Anmerkung                                                                                                                                                                               | ≈ Preis |
|------|----------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------|
| 1×   | Raspberry Pi Zero 2 W                                                      | oder Pi 4/5, falls vorhanden — mehr WLAN- und CPU-Reserve                                                                                                                               | 20 €    |
| 1×   | Pi-Kameramodul v3                                                          | v2 geht auch; muss picamera2 unterstützen. v3 hat Autofokus + besseren Sensor                                                                                                           | 30 €    |
| 1×   | 12-V-Membranpumpe mit Druckschalter (Wohnmobil-Typ, 4–5 bar)               | kräftiger Strahl, selbstansaugend, läuft trocken nicht sofort kaputt. Alternative: Pumpe + Akku aus einer elektrischen Wasserpistole (Taubenturret-Original) — schwächer, aber billiger | 22 €    |
| 1×   | Wasserkarister 2–5 l, flach                                                | muss durch die Basis-Öffnung passen — vorher messen (Abschnitt 04)                                                                                                                      | 8 €     |
| 2 m  | PU-Schlauch 6×4 mm + Rückschlagventil + Messing-Hartstrahldüse Ø0,8–1,2 mm | Düse aus v4 weiterverwenden, falls adaptierbar                                                                                                                                          | 15 €    |
| 2×   | Kugellager 8×22×7 mm (608-Nachbar, wie Taubenturret)                       | Pan-Lagerung des Geschützkopfs                                                                                                                                                          | 4 €     |
| 2 kg | ASA-Filament                                                               | Pflicht bei Sonnenstandort — PLA/PETG verformt sich nachweislich (Taubenturret-Erfahrung)                                                                                               | 45 €    |
| 1×   | (Optional) USB-5-GHz-WLAN-Adapter, Pi-kompatibel                           | drückt die Backend-Latenz deutlich; Kompatibilität vor Kauf prüfen                                                                                                                      | 15 €    |
| 1×   | (Optional) Acrylglas-Fenster Ø40 mm + Silikon                              | Regenschutz vor der Kameraöffnung                                                                                                                                                       | 5 €     |

**Gesamt v5: ~230 €** (davon ~45 € Filament). Wer die 3D-Modelle anpassen will: Der Taubenturret-Autor hat die .f3d-Quelldateien (Fusion 360) veröffentlicht — nach eigener Aussage als CAD-Anfänger, also Struktur nicht erwarten.

### 03 X-10 Tower: Analyse, Skalierung, Umbau, Druck NEU IN v5

Die mitgelieferte STL (*X-10\_Dice\_Tower.stl*, 141.152 Dreiecke) ist ein Würfelturm in Form des X-10-Turbolasers. Vermessene Geometrie im Originalmaßstab:

| Zone          | Höhe (100 %) | max. Radius         | Bedeutung                                         |
|---------------|--------------|---------------------|---------------------------------------------------|
| Basis         | 0–30 mm      | 94 mm (Ø ~190 mm)   | Standfuß, innen hohl → Tank & Elektronik          |
| Schaft        | 30–150 mm    | 65 → 38 mm, konisch | Kabel- und Schlauchkanal (Würfelrutsche entfällt) |
| Geschütztkopf | 150–175 mm   | 75 mm (Ø ~150 mm)   | künftig drehbar: Kamera, Tilt-Servo, Düse         |

#### Skalierung: 200 % ist das Minimum

Im Originalmaßstab ist der Kopf Ø150 × 25 mm — da passt nicht einmal ein MG996R (40×20×43 mm) liegend hinein.

**Empfehlung: 200 % in allen Achsen** → Turm 340 × 340 × 350 mm, Kopf Ø ~300 mm, Basis Ø ~380 mm. Damit passen Tilt-Servo, Kamera und Düse in den Kopf, und die Basis fasst Tank + Pi + Netzteile. Auf einem Bambu X1C/P1S (256 mm Bett) heißt das: **Turm in Segmente schneiden** (Basis, Schaft unten/oben, Kopf) — was ohnehin nötig ist, siehe Umbau.

#### Umbau von Würfelturm auf Geschützturm

Der Dice Tower ist ein statisches Deko-Objekt — für Pan/Tilt sind vier Eingriffe nötig (im Slicer bzw. in Fusion, die STL lässt sich dort per Mesh-Schnitt trennen):

| # | Eingriff                | Umsetzung                                                                                                                                                                                                                                                                      |
|---|-------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1 | Kopf vom Schaft trennen | Planschnitt bei z = 300 mm (skaliert). Schnittflächen bekommen je eine gedruckte Adapterplatte                                                                                                                                                                                 |
| 2 | Pan-Drehlager einbauen  | Drehteller-Prinzip aus dem Taubenturret übernehmen: Pan-Servo sitzt oben im Schaft, treibt über Servohorn/Drive-Adapter den Kopf; die zwei 8×22×7-Kugellager führen die Achse. Die .f3d-Teile (turntable, drive adapter, axlemounts) sind die Vorlage — auf Ø300-Kopf anpassen |
| 3 | Tilt-Mechanik im Kopf   | Tilt-Servo im Kopf (Prinzip servo mount / axlemount left+right aus dem Taubenturret), schwenkt eine Innenlafette mit Düse. Die X-10-Zwillingsrohre bleiben Kosmetik: Düse feuert durch das rechte Rohr, Tilt-Bereich ca. -10°...+45°                                           |
| 4 | Kamerafenster           | Ø40-mm-Öffnung mittig zwischen/über den Rohren, Kamera auf der Tilt-Lafette direkt neben der Düse montieren → minimale Parallaxe. Acrylfenster mit Silikon einsetzen, Shutter-Idee aus dem Taubenturret (camera shutter) optional                                              |

#### Druckempfehlungen

Direkt aus der Taubenturret-Praxis übernommen und für den skalierten Turm verschärft:

| Thema      | Empfehlung                                                                                                                                                                                                                                                                                                          |
|------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Material   | <b>ASA zwingend</b> bei Sonnenstandort (Terrasse/CUBE-Dach). PLA und PETG haben sich beim Original-Taubenturret über die Zeit nachweislich verformt; ASA bringt die nötige UV- und Hitzebeständigkeit. Farbe: hellgrau statt schwarz — senkt die Innentemperatur deutlich (v4-Problem: 70 °C+ im schwarzen Gehäuse) |
| Wandstärke | Beim Skalieren auf 200 % skalieren die konstruierten Wandstärken mit — trotzdem 4–6 Perimeter und 15–20 % Gyroid für Basis und Schaft: der Turm trägt 2–5 kg Wasser und muss Windlast am Geländer aushalten                                                                                                         |
| Toleranzen | Disclaimer des Taubenturret-Autors ernst nehmen: Die Modelle stammen von einem unkalibrierten Drucker; auf dem Bambu ggf. leicht schleifen oder Presspassungen um 0,1–0,2 mm skalieren. Erst Lager- und Servositze als Testschnitte drucken, dann 60 h in den Kopf investieren                                      |

| Thema         | Empfehlung                                                                                                                                                                    |
|---------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Segmentierung | Basis / Schaft (2 Teile) / Kopf. Verbindung über M4-Einschmelzmutter + gedruckte Flansche; Schaftsegmente zusätzlich verkleben (ASA lässt sich mit Acetonschlicker schweißen) |

## 04 Wassertank, Pumpe & Wasserweg NEU IN v5

Der große Systemwechsel: **kein Außenhahn, kein Magnetventil**. Der Turm versorgt sich aus einem Tank in der Basis — genau wie vom Nutzerwunsch vorgesehen („unten kommt der Wassertank rein“).

Tank (2-5 l, Basis) -> Ansaugschlauch mit Filter/Fusssieb  
 -> 12-V-Membranpumpe (Druckschalter, 4-5 bar)  
 -> Rückschlagventil -> PU-Schlauch 6x4 mm durch den Schaft  
 -> Serviceschleife unter dem Drehlager (Pan +/-90° ohne Knick)  
 -> Hartstrahldüse im rechten Turbolaser-Rohr

| Thema              | Detail                                                                                                                                                                                                                                                                                                                                                                                               |
|--------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Tankgröße          | Bei 200 % hat die Basis innen grob Ø300 × 90 mm nutzbar → flacher 3-l-Kanister oder maßgedruckter Tank (ASA ist wasserdicht druckbar, aber ein gekaufter Kanister ist dichter und schneller). Einfüllen über eine Serviceklappe in der Basis-Rückwand                                                                                                                                                |
| Pumpe statt Ventil | Die Membranpumpe mit integriertem Druckschalter hält die Leitung auf Druck; das Relais schaltet die Pumpe direkt (600-ms-Bursts wie in v4). Das Rückschlagventil verhindert Leerlaufen der Steigleitung — sonst spuckt der erste Schuss nur Luft (bekanntes Taubenturret-Problem: Luft in der Kammer → regelmäßiger Spülschuss als Workaround, lässt sich auch in v5 als Cron auf dem Pi einrichten) |
| Reichweite         | 4–5 bar auf Ø1,0-mm-Düse ergibt realistisch 5–8 m harten Strahl — mehr als die alte Magnetventil-Lösung am Hausdruck hinter 2 m ½"-Schlauch                                                                                                                                                                                                                                                          |
| Füllstand          | Nice-to-have: kapazitiver Füllstandssensor (XKC-Y25) außen an den Kanister, GPIO an den Pi → HA-Entity „Tank leer“. Verhindert Trockenlauf-Dauerbetrieb                                                                                                                                                                                                                                              |
| Alternative        | Wer beim Wasserpistolen-Prinzip des Taubenturret bleibt: Pumpe + 7,4-V-Akku aus der Pistole ausschlachten, Relais schaltet Akku+ auf Pumpe (Original-Verkabelung Abschnitt 3.3/3.4 der Anleitung). Nachteil: schwächerer Strahl, Akku alle 2–3 Tage laden                                                                                                                                            |

### ■ REGEN & ELEKTRONIK

Wichtigste Warnung aus der Taubenturret-Anleitung, gilt unverändert: **Der Turm schießt Wasser, ist aber nicht wasserdicht**. Für den Dauer-Außeneinsatz: alle Elektronik auf eine Ebene *über* dem Tank, Kabeldurchführungen nach unten abtropfend, Kamerafenster verkleben, Nähte der Drucksegmente mit Silikon ziehen, und den Kopfspalt (Drehlager) mit einer gedruckten Labyrinth-Schürze überdecken. Bei Starkregen-Prognose: HA-Automation → Servos parken, Abdeckhaube.

## 05 Elektrik & Verkabelung

Verkabelung folgt dem Taubenturret-Schema (GPIO-Nummern identisch), erweitert um Netzbetrieb und Audio-Node. Der Pi wird per Micro-USB aus dem 5-V-Netzteil versorgt — **Netzbetrieb statt Powerbank**: die Powerbank-Lösung des Originals muss alle 2–3 Tage geladen werden, und am CUBE liegt ohnehin Strom.

### Raspberry Pi Zero 2 W

```
GPIO 18 (Pin 12) <-> Pan-Servo Signal (orange)
GPIO 19 (Pin 35) <-> Tilt-Servo Signal (orange)
GPIO 2 (Pin 3) <-> Relais IN (gelb)
3,3 V (Pin 1) <-> Relais VCC (orange)
GND (Pin 6) <-> gemeinsame Masse (5-V- UND 12-V-Netzteil!)
GND (Pin 9) <-> Relais GND (schwarz)
Micro-USB PWR <-> 5-V-Netzteil
```

## Servos, Relais, Pumpe

|                    |     |                       |                     |
|--------------------|-----|-----------------------|---------------------|
| Servos V+ (rot)    | <-> | 5-V-Netzteil (+)      | [nicht aus dem Pi!] |
| Servos GND (braun) | <-> | 5-V-Netzteil (-)      |                     |
| Elko 1000 µF       | <-> | ueber die 5-V-Schiene | (Servo-Einbrueche)  |
| Relais COM         | <-> | 12-V-Netzteil (+)     |                     |
| Relais NO          | <-> | Membranpumpe (+)      |                     |
| Pumpe (-)          | <-> | 12-V-Netzteil (-)     |                     |

### ■ PFLICHT: GEMEINSAME MASSE

Wie in v4: GND von 5-V-Netzteil, 12-V-Netzteil, Pi und Audio-ESP32 verbinden. Der Abwärtswandler aus der Taubenturret-Stückliste entfällt im Netzbetrieb komplett — er war nur nötig, um 5-V-Servos aus dem 7,4-V-Pistolenakku zu speisen (und hatte laut Autor spürbaren Leerlaufverbrauch).

## 06 Software: Taubenturret-Stack + MeLE-Backend NEU IN v5

### 1 · Backend auf dem MeLE Quieter 4C (~20 min)

Das KI-Erkennungs-Backend läuft laut Anleitung auf einem separaten Rechner im Netz — dein Debian-/Docker-Host ist dafür wie gemacht (Pi-hole, Tailscale und HA laufen dort ja bereits):

```
git clone https://github.com/MLWeber/taubenturret-backend
cd taubenturret-backend
# README pruefen: Modell + Port konfigurieren, dann z.B.:
docker build -t taubenturret-backend .
docker run -d --restart unless-stopped -p 8090:8090 taubenturret-backend
```

Merke dir IP:Port (z. B. 192.168.1.20:8090) für die Pi-Konfiguration. Latenzbudget: Bewegung → Snapshot → Backend → Antwort soll unter ~300 ms bleiben, damit die 500-ms-Gesamtreaktion erreichbar ist.

### 2 · Steuerungssoftware auf dem Pi (~40 min)

```
# Raspberry Pi OS Lite (64-bit), Kamera-Ribbon VOR Einbau stecken
git clone https://github.com/MLWeber/taubenturret
# gemaess README installieren; Backend-URL, Servo-Offsets,
# Relais-GPIO und Burst-Dauer in der Konfiguration setzen
```

Funktionsumfang des Stacks laut Projektbeschreibung: kombinierte on-device-Bewegungserkennung + Backend-Objekterkennung, **Web-Dashboard** mit Live-Stream, Clip-Archiv der Bewegungsereignisse und manuellem Ziel-/Feuermodus, sowie **automatische Videoaufzeichnung** bei Bewegung. Das ersetzt die v4-Idee „Kamera irgendwann nachrüsten“ vollständig.

### 3 · MQTT-Brücke zu Home Assistant (~30 min)

Der Taubenturret-Stack bringt von Haus aus keine HA-Integration mit — die rüsten wir mit einem kleinen Python-Wrapper auf dem Pi nach (paho-mqtt, ~40 Zeilen): publiziert *pigeon/shot*, *pigeon/vision\_detected*, *pigeon/tank\_level* und lauscht auf *pigeon/enabled* sowie *pigeon/alert* (Prä-Trigger vom Audio-Node). Alternativ: HA spricht die REST-Endpunkte des Web-Dashboards direkt an. Details je nach Repo-Stand — beim Einrichten aktuelle README lesen, das Projekt bekommt laufend Updates (zuletzt u. a. die .f3d-Dateien, 2026-06).

## 07 Audio-ESP32: neue Rolle

Hardware und Sketch bleiben exakt v4 (inkl. OTA). Nur die Wirkung des MQTT-Topics ändert sich: *pigeon/detected* löst keinen Schuss mehr aus, sondern eine HA-Automation, die den Pi für 60 s in den Alarmmodus versetzt (höhere Framerate/Empfindlichkeit). Damit bleibt die 3-s-Cooldown-Logik unangetastet und der Audio-Kanal kann nie „ins Leere“ feuern.

## 08 Home Assistant

### Entities v5

| Entity                                         | Quelle                         | Bedeutung                |
|------------------------------------------------|--------------------------------|--------------------------|
| camera.pigeon_turret                           | generic/MJPEG vom Pi-Dashboard | Live-Stream im Lovelace  |
| binary_sensor.pigeon_vision_detected           | MQTT (Pi)                      | Kamera-Bestätigung aktiv |
| binary_sensor.pigeon_audio_detected            | MQTT (ESP32)                   | Audio-Kanal (v4-Topic)   |
| binary_sensor.pigeon_audio_online / _pi_online | MQTT LWT                       | Heartbeats beider Nodes  |
| sensor.pigeon_shot_count                       | MQTT (Pi)                      | Schüsse seit Reset       |
| sensor.pigeon_probability                      | MQTT (ESP32)                   | p(Taube) Audio, live     |
| sensor.pigeon_tank_level                       | MQTT (Pi, optional)            | Tank-Füllstand           |

| Entity                                     | Quelle         | Bedeutung                                 |
|--------------------------------------------|----------------|-------------------------------------------|
| input_boolean.pigeon_enabled               | Helper         | Master-Schalter (v4)                      |
| button.pigeon_fire_manual / _park / _flush | MQTT/REST → Pi | Manuell feuern, Servos parken, Spülschuss |

### Automationen (v4-Basis, ergänzt)

• **Nacht-Off / Morgen-On** per Sonnenstand: unverändert übernehmen. • **Audio-Prä-Trigger**: pigeon/detected → REST an Pi (Alarmmodus 60 s). • **Snapshot-Archiv**: bei jedem vision\_detected camera.snapshot — ergänzt das Clip-Archiv des Pi-Dashboards. • **False-Positive-Wächter**: > 50 Schüsse/Tag → Push (v4). • **Tank-Warnung**: Füllstand niedrig → Push + pigeon\_enabled off (Pumpen-Trockenlaufschutz). • **Regen-Park**: Regensensor deiner Ecowitt WS90 meldet Niederschlag → Servos in Parkposition. Der v4-Trend-Tipp gilt weiter: Schuss-Frequenz über 2–3 Wochen plotten — das ist der harte Wirksamkeitsindikator.



## 09 Kalibrierung

| Schritt          | Vorgehen                                                                                                                                                                                                                                                     |
|------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Servo-Nullpunkte | Wie v4: vor dem Aufschrauben der Hörner beide Servos auf Mitte fahren (jetzt per Taubenturret-Webinterface statt Test-Sketch)                                                                                                                                |
| Pan/Tilt-Offsets | Manueller Feuermodus im Web-Dashboard: auf Testziel (Pappteller) in 3 m zielen, Offsets in der Konfiguration nachziehen, bis Strahl und Fadenkreuz deckungsgleich sind                                                                                       |
| Parallaxe        | Kamera und Düse sitzen in v5 bewusst dicht beieinander auf der Tilt-Lafette — trotzdem gilt die Taubenturret-Faustregel: <b>optimal sind 2–5 m Zieldistanz</b> ; darunter schießt der Turm systematisch vorbei, weil der Höhenversatz nicht kompensiert wird |
| Trockentest      | Wie v4: erst ohne Wasser — das Relais-Klicken ist absichtlich hörbares Feedback (Taubenturret-Autor nutzt genau deshalb Relais statt MOSFET). Erst wenn Zielen + Klick sauber sitzen: Tank füllen                                                            |
| Sichtfeld        | Turm so ausrichten, dass die Hauptsitzplätze (Brüstung, Pergola-Kante) im mittleren Bilddrittel liegen; Bildränder haben die größten Winkel-Fehler                                                                                                           |

## 10 Betrieb & Realismus

| Thema         | Maßnahme                                                                                                                                                                                                                             |
|---------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Frost         | v4-Routine angepasst: Tank raus/leeren, Spülschuss-Button bis Leitung leer, Pumpe trocken, pigeon_enabled off. Pan/Tilt-Kopf kann dank ASA draußen bleiben, Elektronik-Segment ist steckbar                                          |
| Sommer        | ASA hält Hitze/UV, aber die Elektronik nicht: heller Druck, Lüftungsschlitze in der Basis-Rückseite (Nordseite), optional 5-V-Lüfter mit Temp-Sensor am Pi → HA                                                                      |
| WLAN          | Der Pi Zero 2 W ist laut Taubenturret-Anleitung der Latenz-Flaschenhals — bei > 1 s Reaktionszeit: 5-GHz-USB-Adapter oder gleich Pi 4                                                                                                |
| Erwartung     | v4-Kurve gilt weiter: Tag 1–3 hohe Trefferzahl, ab Woche 3 idealerweise < 20/Tag. Neu: dank Video-Bestätigung sind die Zahlen jetzt echte Tauben, keine Marder-Phantome                                                              |
| Eskalation    | Wenn auch v5 nicht reicht, bleibt v4-Fazit: passive Maßnahmen (Spikes/Drähte) an Pergola, CUBE und Brüstung — der Hof wird Durchflugkorridor statt Sitzplatz                                                                         |
| Ethik & Recht | Wasser vergrämt, verletzt nicht — bleibt so. Strahl nie auf Nester mit Jungtieren, Nachbargrundstücke oder Personen; Kamera-Sichtfeld auf das eigene Grundstück begrenzen (DSGVO: keine Aufzeichnung von öffentlichem Raum/Nachbarn) |

### FAZIT

v5 vereint das Beste aus drei Welten: die Audio-Früherkennung aus v3/v4, die kamerabasierte Ziel-Lokalisierung des Taubenturret-Projekts und ein Gehäuse, das dem Anlass angemessen ist. Die Tauben von Fürth werden dem Imperium nicht entkommen — und falls doch, hast du wenigstens einen sehr guten Würfelturm gedruckt.