# 🚀 NETRAX AI - 5-Minute Quickstart

Get NETRAX running in **5 minutes or less**.

---

## ⚡ Super Quick Start (Docker)

```bash
# 1. Download files
git clone <repository>
cd netrax-vision

# 2. Configure
cp .env.example .env

# 3. Run
docker-compose up -d

# 4. Open frontend
# Open index.html in your browser
```

**Done!** 🎉

- Backend: http://localhost:8000
- WebSocket: ws://localhost:8000/ws
- Video Feed: http://localhost:8000/video_feed

---

## 🐍 Manual Setup (No Docker)

```bash
```

```
# 1. Install dependencies
pip install -r requirements.txt

# 2. Configure
cp .env.example .env

# 3. Run server
python main.py

# 4. Open frontend
# Open index.html in your browser
```

**Done!** 🎉

---

## 📁 Minimal File Structure

You need these files:

```
netrax-vision/
├── main.py                 # ✅ Core server
├── config.py               # ✅ Configuration
├── requirements.txt        # ✅ Dependencies
├── .env                    # ✅ Your settings
├── vision_engine/
│   ├── __init__.py         # ✅ (empty file)
│   ├── body_tracker.py     # ✅ Body tracking
│   ├── iris_tracker.py     # ✅ Eye tracking
│   ├── gesture_engine.py   # ✅ Gestures
│   ├── object_detector.py  # ✅ Objects (optional)
│   ├── tracking_coordinator.py # ✅ Orchestrator
│   ├── visualizer.py       # ✅ Visuals
│   └── filters.py          # ✅ Kalman filters
├── frontend/
│   └── index.html          # ✅ Your UI
└── models/                 # 📁 (auto-created)
```

---

## 🔧 Minimum .env Configuration

bash

```
# Camera
CAMERA_INDEX=0

# Modules (disable if slow)
ENABLE_BODY_TRACKING=true
ENABLE_IRIS_TRACKING=true
ENABLE_GESTURE_RECOGNITION=true
ENABLE_OBJECT_DETECTION=false  # Heavy!

# Performance
USE_GPU=false  # Set true if you have GPU
TARGET_FPS=30
```

---

## 🧪 Test It Works

### 1. Check Server

bash

```
curl http://localhost:8000/status
```

Should return:

json

```
{
  "camera_active": true,
  "active_connections": 0,
  "vision_coordinator": true
}
```

### 2. Test WebSocket

Open browser console and run:

javascript

```
const ws = new WebSocket('ws://localhost:8000/ws');
ws.onmessage = (e) => console.log(JSON.parse(e.data));
```

You should see messages like:

json

```
{"type": "stats", "stats": {"fps": 30, ...}}
```

### 3. View Video Feed

Open: http://localhost:8000/video_feed

You should see your camera with overlays.

---

## 🎯 Common Issues & Fixes

### Camera Not Found

```bash
# Linux - check devices
ls /dev/video*

# Test camera
python -c "import cv2; print(cv2.VideoCapture(0).isOpened())"

# Fix: Update CAMERA_INDEX in .env
```

### ImportError: mediapipe

```bash
pip install mediapipe==0.10.8
```

### Slow Performance

```bash
# In .env, reduce resolution:
FRAME_WIDTH=640
FRAME_HEIGHT=480

# Or disable heavy modules:
ENABLE_OBJECT_DETECTION=false
BODY_MODEL_COMPLEXITY=0
```

### WebSocket Won't Connect

```bash
```

```
# Check server is running
curl http://localhost:8000/

# Check firewall
sudo ufw allow 8000  # Linux
```

---

## Frontend Integration (Minimal)

html

```
<!DOCTYPE html>
<html>
<head>
    <title>NETRAX Test</title>
</head>
<body>
    <!-- Video feed -->
    <img id="video" src="http://localhost:8000/video_feed"
         style="width: 640px;">

    <!-- Stats -->
    <div id="stats"></div>

    <script>
        // Connect WebSocket
        const ws = new WebSocket('ws://localhost:8000/ws');

        ws.onmessage = (event) => {
            const data = JSON.parse(event.data);

            if (data.type === 'stats') {
                document.getElementById('stats').innerHTML =
                    `FPS: ${data.stats.fps.toFixed(1)}`;
            }

            if (data.type === 'gesture_command') {
                console.log('Gesture:', data.command);
                alert(`Detected: ${data.command}`);
            }
        };
    </script>
</body>
</html>
```

Save as `test.html` and open in browser!

---

## 🎮 Try These Gestures

1. **Peace Sign** ✌️ - Index and middle fingers up
2. **Stop** 🖐️ - Open palm
3. **Thumbs Up** 👍 - Thumb up, other fingers down
4. **Fist** ✊ - Closed hand

Move in front of camera and watch console!

---

## 🔥 Next Steps

1. ✅ **Working?** Read full README.md
2. ✅ **Customize:** Edit `.env` settings
3. ✅ **Integrate:** Connect to your app
4. ✅ **Deploy:** Use Docker for production
5. ✅ **Extend:** Add custom gestures

---

## 📚 Full Documentation

- **Setup:** README.md
- **Architecture:** ARCHITECTURE.md
- **API Docs:** Check `/api/gestures` endpoint

---

## 🆘 Still Having Issues?

1. **Check logs:**

```bash
```

```
# Docker
docker-compose logs -f

# Manual
python main.py  # Watch console output
```

2. **Test camera:**

```bash
python -c "import cv2; cv2.VideoCapture(0).read()[1]"
```

3. **Test MediaPipe:**

```bash
python -c "import mediapipe; print('OK')"
```

4. **Read troubleshooting:** README.md#troubleshooting

---

## 💡 Pro Tips

- **Performance:** Lower resolution for higher FPS
- **GPU:** Massive speed boost if available
- **Object Detection:** Disable unless needed (very heavy)
- **Kalman Filters:** Enable for smoother tracking
- **Debug Mode:** Set `DEBUG=true` for verbose logs

---

**You're ready! NETRAX is watching.** 👁️

Questions? Check the full documentation or open an issue.