**VIOLENCETRACKER PRO - ADVANCED VIOLENCE DETECTION SYSTEM**

Advanced AI system for real-time violence detection using both reactive and proactive analysis methods with predictive capabilities.

## I. OVERVIEW

ViolenceTracker Pro is a sophisticated computer vision system that combines:

- **Reactive Analysis**: Traditional violence detection in current frames
- **Proactive Analysis**: Predictive forecasting of potential violence escalation
- **Real-time Comparison**: Live visualization of both methods' performance

## II. DEMO

https://docs/images/interface_demo.png ← *Add actual screenshot*

## III.KEY FEATURES

### Dual Analysis Modes

- **Reactive Detection**: Analyzes current frame sequences for violence
- **Proactive Prediction**: Forecasts potential violence 5-8 seconds ahead

### Advanced GUI

- Real-time video display with overlay alerts
- Interactive comparison charts
- Progress tracking and detailed console output
- Advantage statistics and performance metrics

### Smart Alerts

- **GREEN**: Normal situation (STABLE SITUATION)
- **ORANGE**: Early warning (ESCALATION LIKELY)

- **RED**: High danger (CONFLICT CONTINUES, DANGER!)

**IV.METHODOLOGY**

**Reactive Analysis**

```
# Traditional frame-by-frame analysis
input_sequence = frames[-16:]  # Last 16 frames
violence_probability = model.predict(input_sequence)
```

**Proactive Analysis**

```
# Predictive escalation detection
motion_trend = analyze_motion_trend(sequence)      # Movement patterns
intensity_trend = analyze_intensity_trend(sequence) # Change intensity
predicted_prob = current_prob + motion_trend * 0.3 + intensity_trend * 0.2
```

**V. INSTALLATION**

**Prerequisites**

- Python 3.8+
- TensorFlow 2.x
- OpenCV
- PyQt5

**Step-by-Step Setup**

1. **Clone Repository**

```
git clone https://github.com/your-username/violencetracker-pro.git
cd violencetracker-pro
```

2. **Create Virtual Environment**

```
python -m venv violence_env
```

source violence_env/bin/activate  # *Linux/Mac*

violence_env\Scripts\activate  # *Windows*

3. **Install Dependencies**

   pip install -r requirements.txt

4. **Download Pre-trained Model**

   # *Place your model in project root*

   # *Model should be named: best_model_Unidirectional_LSTM.h5*

**VI. REQUIRED DEPENDENCIES**

Create requirements.txt:

tensorflow>=2.8.0

opencv-python>=4.5.0

PyQt5>=5.15.0

numpy>=1.21.0

pandas>=1.3.0

scipy>=1.7.0

**VII.USAGE**

**Starting the Application**

python main.py

**Basic Workflow**

1. **Launch Application**
   - System loads the AI model automatically
   - Main interface with video panel and analytics displays
2. **Select Analysis Mode**

- Analysis → Advanced Analysis (Both Methods): Full dual analysis
- Analysis → Reactive Analysis Only: Traditional single method
- Demo → Live Advantage Demo: Demonstration mode

3. **Load Video Folder**
   - Select folder containing MP4/AVI/MOV files
   - System processes videos automatically
4. **Monitor Results**
   - Real-time video with overlay alerts
   - Live comparison graphs
   - Console with detailed analysis log

**VIII.EXAMPLE OUTPUT INTERPRETATION**

Reactive: 77.1%,

Proactive: 77.2%,

ALERT: CONFLICT CONTINUES ADVANTAGE: +0.1%, STATUS: DANGER!

- Both methods detect high violence probability (77%)
- Proactive method shows slight advantage (+0.1%)
- Situation is actively dangerous

**Results Interpretation**

**Confidence Levels**

- **0-30%**: Normal situation
- **30-60%**: Potential risk
- **60-80%**: High probability of violence

- **80-100%**: Certain violence detection

### Alert Types

| Alert Level | Color | Meaning | Action |
|---|---|---|---|
| STABLE SITUATION | Green | No violence detected | Monitor |
| ESCALATION LIKELY | Orange | Potential violence developing | Increase vigilance |
| CONFLICT CONTINUES | Red | Active violence occurring | Immediate response |
| DANGER! | Red | Critical violence level | Emergency response |

### Advantage Metrics

- **+0.1% to +5%**: Minor early detection
- **+5% to +15%**: Significant advantage
- **+15%+**: Major proactive benefit

## IX. PROJECT STRUCTURE

violencetracker-pro/

```
├── main.py                    # Main application entry point
├── best_model_Unidirectional_LSTM.h5 # Pre-trained AI model
├── requirements.txt           # Python dependencies
│
├── src/                       # Source code directory
│   ├── advanced_processor.py      # Advanced video processing
│   ├── proactive_predictor.py     # Proactive analysis logic
│   ├── tradiction_predictor.py    # Complex for reactive detection analysis
│   ├── gui_components.py          # UI components and widgets
│   └── utils.py                   # Utility functions
│
├── docs/                      # Documentation
│   ├── images/                    # Screenshots and diagrams
│   └── technical_details.md       # Technical specifications
│
├── examples/                  # Sample videos for testing
│   ├── normal_scenario.mp4
│   └── violence_scenario.mp4
```

```
└── results/                # Analysis output directory
    ├── logs/               # Processing logs
    └── exports/            # Result exports
```

## X.TECHNICAL DETAILS

### AI Architecture

**Base Model**: RLCN-LSTM

- **Input Size**: 64x64 pixels, 16-frame sequences
- **Processing**: Real-time at ~40ms per frame
- **Output**: Violence probability (0-1 scale)

### Performance Metrics

- **Frame Rate**: 25 FPS processing
- **Accuracy**: >85% on test datasets
- **Advantage**: 2-8 seconds early detection
- **Memory**: ~500MB RAM usage

### Supported Formats

- **Video**: MP4, AVI, MOV
- **Resolution**: Adaptive (optimized for 640x480)
- **Framerate**: 15-30 FPS recommended

## XI. CONTRIBUTING

We welcome contributions! Please see our Contributing Guidelines for details.

### Development Setup

```
# Fork and clone the repository
git clone https://github.com/your-username/violencetracker-pro.git
# Create feature branch
git checkout -b feature/amazing-feature
# Commit changes
git commit -m "Add amazing feature"
# Push to branch
git push origin feature/amazing-feature
# Create Pull Request
```

**XII. LICENSE**

This project is licensed under the MIT License - see the LICENSE file for details.

**Support**

- **Email**: support@violencetracker.com
- **Issues**: GitHub Issues
- **Documentation**:

**Acknowledgments**

- TensorFlow team for deep learning framework
- OpenCV community for computer vision tools
- Research papers on violence detection algorithms

**If you find this project useful, please give it a star on GitHub!**

🔄 **Version History**

- **v1.0.0** (2024-01-15)
    - Initial release with reactive/proactive analysis
    - Real-time comparison interface
    - Advanced alert system

*Last Updated: oct 2025*