

# Руководство для студента: Система стабилизации масок

Привет! 

Это руководство поможет тебе понять, что делает система и как её запустить. Написано простым языком, без лишней сложности.

---

## Часть 1: Что это за система и зачем она нужна?

### Проблема, которую мы решаем

Представь, что ты хочешь выделить человека на видео — например, чтобы заменить фон или отследить движение.

Нейросеть может найти человека на каждом кадре и создать **маску** — это как трафарет, который показывает, где находится человек (белое) и где фон (чёрное).

**Но есть проблема:** нейросеть обрабатывает каждый кадр отдельно, не зная про соседние кадры. Из-за этого маска немного "прыгает" между кадрами — это называется **мерцание (flickering)**.

Кадр 1: Мaska чуть левее  
Кадр 2: Мaska чуть правее  
Кадр 3: Мaska снова левее  
→ Получается дрожащий, мерцающий контур 

### Наше решение

Мы применяем **временное сглаживание** — смотрим на несколько соседних кадров и усредняем маску. Это убирает мерцание:

Было:  (прыгает)  
Стало: ● ● ● ● (плавно)

### Как измерить улучшение?

Используем метрику **IoU (Intersection over Union)** — она показывает, насколько похожи маски соседних кадров:

- $\text{IoU} = 1.0 \rightarrow$  маски идентичны (идеально)

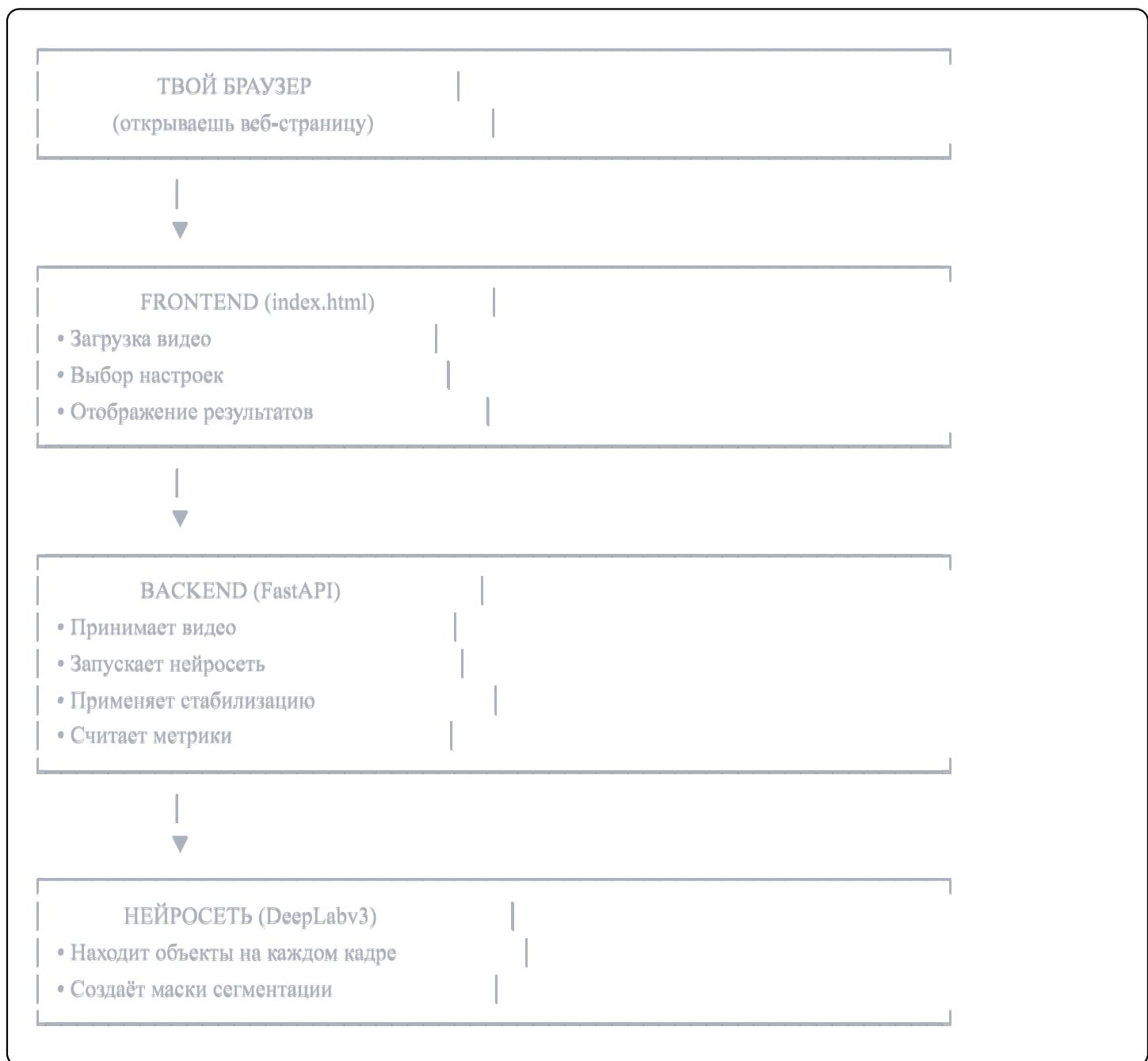
- IoU = 0.5 → маски совпадают наполовину
- IoU = 0.0 → маски совсем разные

**Чем выше IoU — тем стабильнее маска, тем меньше мерцание!**

---

## 💡 Часть 2: Как устроена система?

### Компоненты



## Файловая структура проекта

```
mask-stabilization/
|
|   └── src/          # Код бэкенда (Python)
|       ├── main.py    # API сервер (FastAPI)
|       ├── segmentation.py # Работа с нейросетью
|       ├── stabilization.py # Алгоритмы сглаживания
|       ├── metrics.py   # Расчёт IoU и других метрик
|       └── utils.py     # Вспомогательные функции
|
|   └── frontend/      # Веб-интерфейс
|       └── index.html  # Страница с интерфейсом
|
|   └── docs/          # Документация
|       ├── METHODOLOGY.md # Подробное техническое описание
|       └── QUICKSTART.md # Краткая инструкция
|
|   └── scripts/        # Скрипты настройки
|       └── setup_windows.ps1 # Настройка Windows-сервера
|
|   └── notebooks/      # Jupyter ноутбуки для экспериментов
|       └── analysis.ipynb # Анализ результатов
|
|   └── start_system.sh # 🚀 Скрипт запуска всей системы
|   └── README.md       # Описание проекта
```

## Методы стабилизации

В системе есть 3 метода сглаживания:

Метод	Как работает	Когда использовать
<b>Moving Average</b>	Среднее по N соседним кадрам	Универсальный, начни с него
<b>Median Filter</b>	Медиана по N кадрам	Когда есть резкие выбросы
<b>Exponential Smoothing</b>	Взвешенное среднее (новые кадры важнее)	Когда объект быстро движется

## Где всё работает?

Система работает на сервере с Jupyter (там есть мощный GPU для нейросети). Но из-за ограничений сети мы не можем напрямую открыть порт. Поэтому используем **SSH-туннель** через Windows-компьютер.

```
Твой браузер → Windows-сервер → SSH-туннель → Jupyter-сервер
```

## Запуск (когда всё уже настроено)

1. Открой терминал в Jupyter (на сервере 176.109.92.50)
2. Перейди в папку проекта:

```
bash  
cd /tf/darachubarova/defliker
```

3. Запусти систему одной командой:

```
bash  
./start_system.sh
```

4. Открой в браузере:

```
http://23.189.104.205:8080
```

5. Готово! Можешь загружать видео и экспериментировать.

## Полезные команды

```
bash
```

```
# Проверить статус системы  
./start_system.sh --status  
  
# Остановить всё  
./start_system.sh --stop  
  
# Посмотреть логи (если что-то не работает)  
./start_system.sh --logs  
  
# Показать справку  
./start_system.sh --help
```

## 🎮 Часть 4: Как пользоваться веб-интерфейсом?

### Шаг 1: Загрузка видео

- Открой <http://23.189.104.205:8080>
- Перетащи видеофайл в зону загрузки (или нажми и выбери)
- Подожди, пока видео загрузится

### Шаг 2: Настройка

- **Object Class** — что искать на видео (Person, Car, Dog...)
- **Stabilization Method** — метод сглаживания
- **Window Size** — размер окна (больше = сильнее сглаживание)

### Шаг 3: Запуск

- Нажми "STABILIZE VIDEO"
- Подожди (обработка занимает несколько минут)
- Смотри прогресс на экране

### Шаг 4: Результаты

После обработки увидишь:

- **BEFORE** — маска до стабилизации

- **AFTER** — маска после стабилизации
- **COMPARISON** — сравнение на исходном кадре
- **Метрики** — IoU Before, IoU After, Improvement %

Можешь листать кадры кнопками **Previous / Next**.

---

## **Часть 5: Если что-то не работает**

**Проблема:** Сайт не открывается

**Проверь:**

1. Запущена ли система? `./start_system.sh --status`
2. Если нет — запусти: `./start_system.sh`

**Проблема:** Туннель отключился

**Решение:**

```
bash  
./start_system.sh --stop  
./start_system.sh
```

**Проблема:** Backend не запускается

**Посмотри логи:**

```
bash  
cat /tf/darachubarova/defliker/logs/backend.log
```

**Проблема:** Обработка очень долгая

Это нормально для длинных видео! 918 кадров Full HD = 5-10 минут.

**Совет:** Для тестов используй короткие видео (5-10 секунд).

---

## Часть 6: Что почитать дальше?

Для понимания кода:

1. `src/main.py` — начни отсюда, это главный файл сервера
2. `src/segmentation.py` — как работает нейросеть
3. `src/stabilization.py` — алгоритмы сглаживания (простая математика!)
4. `src/metrics.py` — как считается IoU

Для понимания теории:

1. `docs/METHODOLOGY.md` — подробное техническое описание
2. `notebooks/analysis.ipynb` — эксперименты с визуализацией

Для понимания инфраструктуры:

1. `start_system.sh` — как запускается система
  2. `docs/METHODOLOGY.md` — раздел "Сетевая инфраструктура"
- 

## Советы для изучения

1. **Начни с веб-интерфейса** — поиграйся с разными видео и настройками
  2. **Посмотри результаты** — сравни Before/After, пойми что меняется
  3. **Читай код с `main.py`** — он самый понятный, там комментарии
  4. **Экспериментируй** — меняй параметры, смотри что получается
  5. **Не бойся спрашивать** — если что-то непонятно!
- 

## Контрольные вопросы (проверь себя)

1. Что такое "мерцание маски" и почему оно возникает?
2. Что показывает метрика IoU?
3. Какие три метода стабилизации есть в системе?

4. Зачем нужен SSH-туннель?
  5. Какой командой запустить систему?
  6. Где находится код нейросети?
- 

*Удачи в изучении!* 