🚀 Quick Start Guide

Get Anime Auto-Clipper running in 5 minutes!

Prerequisites

- Docker & Docker Compose installed

- 4GB+ free RAM

- 10GB+ free disk space

Step 1: Clone & Setup (1 minute)

```bash
# Clone the repository
git clone https://github.com/yourusername/anime-auto-clipper.git
cd anime-auto-clipper

# Make setup script executable
chmod +x setup.sh

# Run setup (this does everything for you!)
./setup.sh
```

The setup script will:

- Create all necessary directories

- Generate environment files

- Start Docker containers

- Run database migrations

- Create S3 bucket

Step 2: Verify Services (30 seconds)

Check that all services are running:

```bash
docker-compose ps
```

You should see 7 services running:

- ✅ db (PostgreSQL)

- ✅ redis

- ✅ minio (S3 storage)

- ✅ api (FastAPI backend)

- ✅ worker (Celery worker)

- ✅ flower (Celery monitor)

- ✅ frontend (Next.js)

Step 3: Access the Application (30 seconds)

Open your browser and visit:

Frontend: http://localhost:3000

Other useful URLs:

- **API Docs**: http://localhost:8000/docs

- **MinIO Console**: http://localhost:9001 (minioadmin / minioadmin)

- **Celery Monitor**: http://localhost:5555

Step 4: Upload Your First Video (2 minutes)

1. Click **"Browse Files"** or drag-and-drop an anime video

2. Add optional keywords (e.g., "fight", "epic")

3. Click **"Start Analysis"**

4. Wait 2-5 minutes for AI processing

5. Browse generated clips and select your favorites

6. Choose caption style and export!

🎉 That's It!

You're now ready to create viral anime clips!

Common Commands

View Logs

```bash
# All services
docker-compose logs -f

# Specific service
docker-compose logs -f worker
docker-compose logs -f api
```

Restart a Service

```bash
docker-compose restart worker
docker-compose restart api
```

Stop Everything

```bash
docker-compose down
```

Reset Database

```bash
docker-compose down -v  # Warning: deletes all data!
docker-compose up -d
docker-compose exec api alembic upgrade head
```

Access Database

```bash
docker-compose exec db psql -U clipper -d anime_clipper
```

Troubleshooting

Services won't start

```bash
# Check Docker daemon is running
docker ps

# Check logs for errors
docker-compose logs
```

Port conflicts

If ports are already in use, edit docker-compose.yml:

```yaml
# Change these ports to available ones
- "3000:3000"  # Frontend
- "8000:8000"  # API
- "5432:5432"  # Database
```

Worker not processing jobs

```bash
# Check worker logs
docker-compose logs -f worker

# Restart worker
docker-compose restart worker

# Check Celery monitor
open http://localhost:5555
```

Out of disk space

```bash
```

```bash
# Clean up old Docker resources
docker system prune -a --volumes

# Remove old video files
docker-compose exec api rm -rf /tmp/videos/*
```

FFmpeg errors

Make sure the video file is a supported format:

- MP4 (H.264)

- MKV

- AVI

- MOV

Whisper model download

First run downloads Whisper model (~1GB). This is normal.

Development Mode

Backend Development

```bash
cd backend
python -m venv venv
source venv/bin/activate  # Windows: venv\Scripts\activate
pip install -r requirements.txt
uvicorn app.main:app --reload --port 8000
```

Frontend Development

```bash
cd frontend
npm install
npm run dev
```

Run Tests

```bash
# Backend tests
cd backend
pytest tests/ -v

# Frontend tests
cd frontend
npm test
```

Production Deployment

Environment Variables

Create `.env.production`:

```bash
DATABASE_URL=postgresql://user:pass@prod-db:5432/anime_clipper
REDIS_URL=redis://prod-redis:6379/0
S3_ENDPOINT=https://s3.amazonaws.com
S3_ACCESS_KEY=your_key
S3_SECRET_KEY=your_secret
JWT_SECRET=your_secure_random_string
```

Docker Compose Production

```bash
docker-compose -f docker-compose.prod.yml up -d
```

Scale Workers

```bash
docker-compose up -d --scale worker=4
```

Next Steps

1. **Customize Configuration**: Edit `config.yaml` to adjust clip length, scoring weights, etc.

2. **Add GPU Support**: Edit `docker-compose.yml` to enable GPU for faster processing

3. **Set Up Monitoring**: Connect to external logging (Sentry, DataDog, etc.)

4. **Configure CDN**: Use CloudFront or similar for faster video delivery

5. **Add Authentication**: Integrate OAuth2 providers (Google, GitHub, etc.)

Support

- **Documentation**: See `README.md`

- **Issues**: https://github.com/yourusername/anime-auto-clipper/issues

- **Discord**: https://discord.gg/animeclipper

Happy clipping! 🎬 ✂️