**Application**

A web-based application that allows users to upload images of handwritten digits, processes the images through preprocessing steps, and uses a trained SVM model to predict the digit.

---

**Architecture Selected: Microservices**

**Explanation:**

The application uses a microservices architecture to modularize key functional units:

- **UI Service**: Front-end for image upload and result display.
- **Preprocessing Service**: Applies noise reduction, resizing, normalization, and contour extraction.
- **Model Inference Service**: Loads the trained SVM model and returns predictions.
- **Storage Service**: Handles data storage for user input and logs.

This separation allows easy scaling, independent development, and faster iterations.

---

**Diagrams to be Created (Descriptions below):**

- **Use Case Diagram**: Shows interaction between user and system (upload image → receive prediction).
- **Class Diagram**: Includes classes like `DigitImage`, `Preprocessor`, `SVMModel`, etc.
- **DFD**: Data flow from image input through processing to classification and result output.
- **Component Diagram**: Microservices and their interactions.
- **Sequence Diagram**: Upload → Preprocess → Predict → Display result.
- **Deployment Diagram**: Services deployed across containers (e.g., Docker), with S3/Blob storage backend.

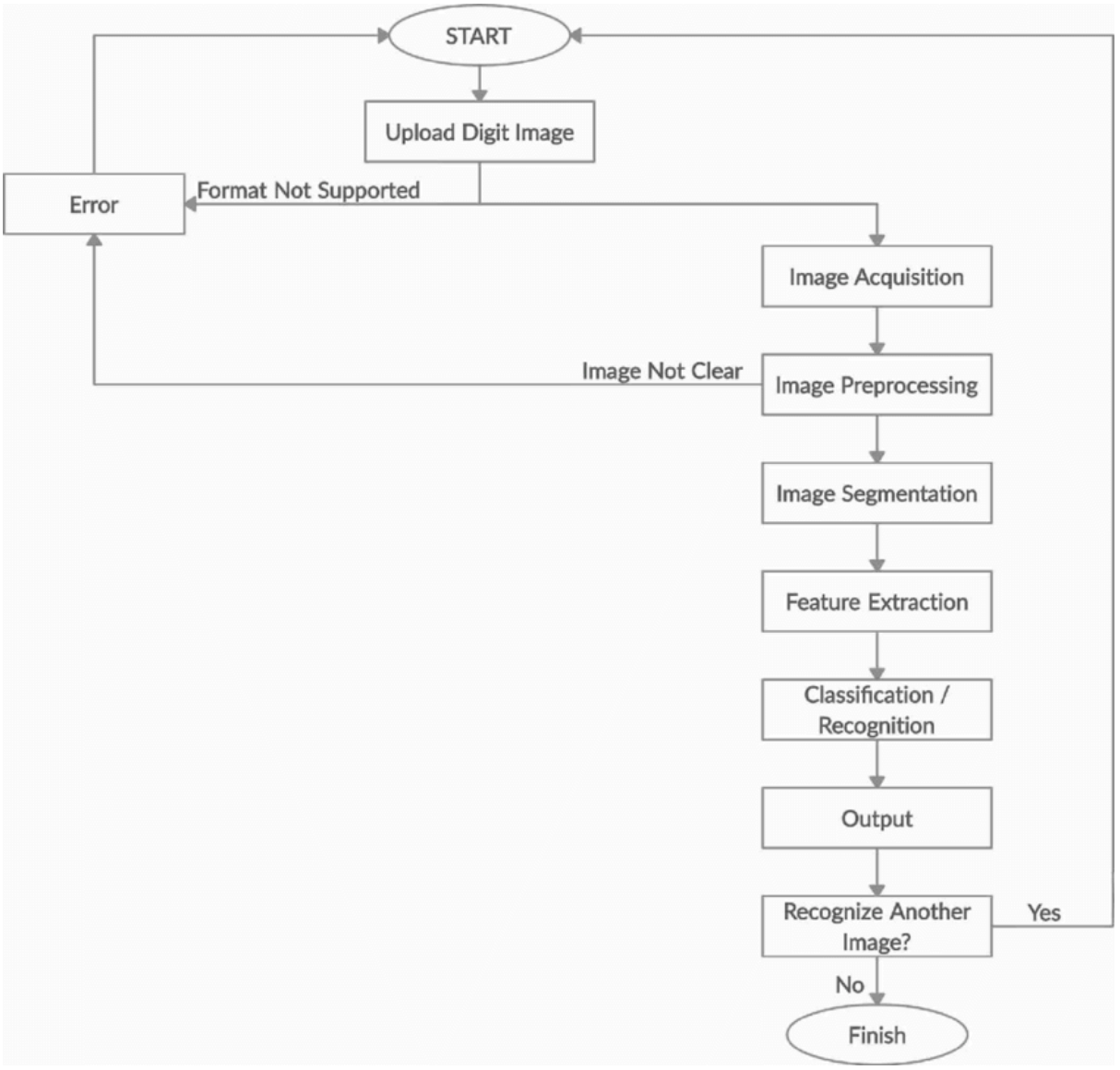
---

**Database**

Used to store user-uploaded images, preprocessing logs, and prediction results for analysis and audit purposes.

---

**ER Diagram**



---

**Data Sets**

- MNIST dataset used for training.
- User-provided images for prediction.