5. MLOPS REPORT

Model Deployment and Monitoring Infrastructure

Deployment Architecture

API Design

```
Framework: FastAPI
Endpoints: /predict, /health, /metrics
Input: Base64 encoded images
Output: JSON predictions with confidence scores
```

Infrastructure

- Containerization: Docker with optimized base image
- Orchestration: Kubernetes for scaling
- Monitoring: Prometheus + Grafana dashboard
- Logging: Structured JSON logging

Model Serving Pipeline

- Preprocessing Service
- Image decoding and validation
- Standardization (64x64 resize)
- Normalization (ImageNet statistics)
- Batch processing support

Prediction Service

- Model loading and warm-up
- GPU acceleration support
- Request queuing and load balancing
- Response caching for performance

Monitoring Setup

Performance Metrics

- Latency: P95 < 100ms
- Throughput: 100+ requests/second
- Accuracy: Continuous validation

- Data Drift: Statistical monitoring

Alerting System

Performance degradation alerts

- Data drift detection
- Resource utilization monitoring
- Error rate thresholds

Retraining Strategy

Trigger Conditions

- Accuracy drop >5% over 7 days
- Significant data drift detected
- New land type categories added
- Quarterly scheduled retraining

Pipeline Automation

- Data Collection: New image ingestion
- Validation: Data quality checks
- Retraining: Automated model updates
- Testing: Performance validation
- Deployment: Canary release strategy

Scalability Considerations

- Horizontal scaling for high throughput
- Regional deployment for low latency
- Cost optimization with spot instances
- Cold start mitigation strategies