

Lens Tagger

The Problem

One can't search in the Photos app by what lens was used (0.5x, 1x, 2x etc)

Project Overview

A macOS app that makes iPhone camera lens types (and potentially other EXIF data) searchable in the Photos app by adding keywords and optional albums.

Core Features

- Read lens type from photo metadata
- Add searchable keywords to Photos library
- Create optional albums by lens type
- Show live results in Photos app
- Handle both iPhone-specific and general photography terms

User Interaction Flows

Input Methods

1. Direct Launch
 - Open app
 - Present system photo picker
 - Process selected photos
2. Photos Integration
 - Drag photos from Photos app

- Share extension
- Services menu integration
- Process received photos

Configuration Dialog

- Shows only lens types found in selection
- Columns:
 - Lens Type (0.5x, 1x, etc)
 - Tag Name (editable)
 - Create Album (checkbox)
 - Count (total/unprocessed)
- Default tag names provided
- Live preview of results

Technical Components

Photos Framework Integration

- PHPhotoLibrary authorization
- PHAsset management
- Album creation
- Keyword management
- Live updates to Photos app

EXIF Processing

```
enum LensCategory {
    case ultraWide      // <= 16mm
    case wide           // ~16-35mm
    case standard       // ~35-70mm
    case telephoto      // ~70-300mm
    case superTele      // > 300mm
}

// Extensible metadata handling
protocol MetadataExtractor {
    var tagName: String { get }
    func extract(from asset: PHAsset) async throws -> String?
}
```

Data Model

```
struct LensStats {
    let multiplier: String
    let focalLength: Double
    let totalCount: Int
    let unprocessedCount: Int
    var tagName: String
    var createAlbum: Bool
}
```

Processing Flow

1. Receive photos (picker or drag)
2. Read EXIF data
3. Group by lens type
4. Present configuration
5. Process user choices:
 - Add keywords
 - Create albums
 - Track processed state
6. Show results

Future Expansion Possibilities

- Support for additional EXIF tags
- DSLR/mirrorless camera support
- Batch processing features
- Custom tag naming templates
- Export/import of tag configurations
- Integration with other photo management apps

Technical Considerations

- Asynchronous processing for large sets
- Error handling and recovery
- iCloud Photos compatibility
- Progress reporting
- Version tracking for processed photos

User Experience Goals

- Simple, focused interface
- Immediate feedback in Photos app
- Familiar terminology for iPhone users
- Professional terminology for DSLR users
- Clear indication of processing status
- Easy recovery from errors

Notes

- Keeps Photos as the source of truth
- Uses native Photos framework APIs
- Maintains compatibility with Photos search
- Preserves existing metadata
- Handles future iPhone models without hardcoding

Development Priorities

1. Core EXIF reading and keyword tagging
2. Basic configuration UI
3. Photos app integration

4. Additional input methods
5. Enhanced filtering and organization
6. Extended metadata support