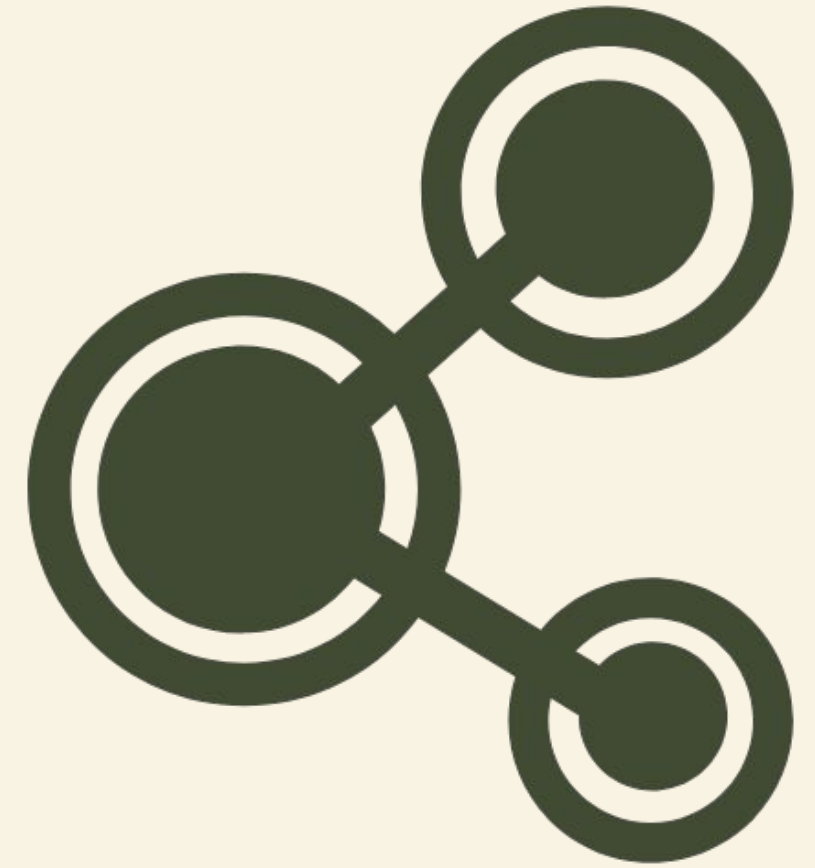


Building Efficient and Frugal Image Data Processing Pipelines



- **Image Processing Use Cases**
- **Challenges**
- **The State of Art**
- **Future**



Introduction - The Importance of Image Data



Healthcare

(medical imaging,
diagnosis)



Autonomous vehicles

(perception, object
recognition)



E-commerce

(product images, visual
search)



Social media

(content sharing, facial
recognition)

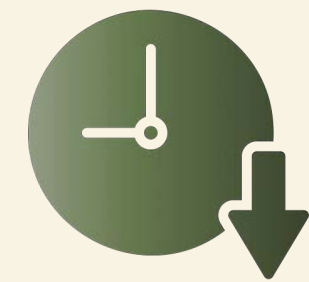
...and many more!

LARGE IMAGE DATASETS POSE SIGNIFICANT CHALLENGES:

- High storage costs
- Intensive computational requirements
- Long processing times
- Network bandwidth limitations



Goals of Efficient and Frugal Pipelines



Minimize processing time



Reduce storage costs

OPTIMIZE RESOURCE UTILIZATION

Choosing the Right Image Formats - Image Formats



JOINT PHOTOGRAPHIC EXPERTS GROUP



Lossy compression format widely used for photographs and images with complex color gradients.

Offers a good balance between file size and image quality.

Supports adjustable compression levels, allowing for fine-grained control over file size and quality trade-offs.

Widely supported by web browsers, image editing software, and devices.

PORTABLE NETWORK GRAPHICS



Lossless compression format suitable for images with sharp edges, text, and transparency.

Preserves image quality even after multiple edits.

Supports a wider range of colors and transparency options than JPEG.

Often used for logos, icons, and graphics.

Image Formats: WebP, AVIF, HEIC – The Next Generation



- Developed by Google, offers both lossy and lossless compression;
- Provides superior compression compared to JPEG and PNG;
- Gaining popularity for web images due to smaller file sizes and faster loading times.



- Based on the AV1 video codec, promising even better compression than WebP and HEIC;
- Still in early stages of adoption but has potential for future web image optimization.




- Developed by Apple, uses HEVC (High Efficiency Video Coding) for compression;
- Offers better compression than JPEG while maintaining high image quality;
- Ideal for storing photos on devices with limited storage space.



LOSSY COMPRESSION

- Achieves higher compression ratios;
- May result in some loss of image quality;
- Suitable for applications where minor; quality loss is acceptable.

LOSSLESS COMPRESSION

- Preserves all image data;
 - Results in larger file sizes;
 - Ideal for applications where image; fidelity is critical.
- 

Compression Libraries and Tools

Compression libraries and tools provide pre-built functionalities for encoding and decoding various image formats, enabling efficient compression and decompression of images within your applications.

They often include optimized algorithms and hardware acceleration capabilities, leading to significant performance improvements compared to custom implementations.

These libraries simplify the image handling process, allowing you to focus on core application logic rather than intricate compression algorithms.

libjpeg-
turbo



ADAPTIVE COMPRESSION

Adaptive compression is a technique where compression levels are dynamically adjusted based on the content and quality requirements of an image.

BENEFITS

- Optimized Storage and Bandwidth;
- Improved User Experience;
- Cost Savings.

how it works?

IMAGE ANALYSIS

The compression algorithm analyzes the image content to identify areas with varying levels of detail and complexity

COMPRESSION LEVEL ADJUSTMENT

- Compression levels are adjusted dynamically for different regions of the image.
- Areas with less detail (e.g., smooth backgrounds) can be compressed more aggressively.
 - Areas with more detail (e.g., faces, edges) are compressed less to preserve visual quality.

ENCODED IMAGE

The final encoded image has a smaller file size while maintaining acceptable visual quality.

Leveraging Cloud-Native Services – Cloud Storage for Scalable Image Storage

Cloud computing offers a wealth of services that can significantly enhance the efficiency and cost-effectiveness of your image processing pipelines.



Cloud storage services like Amazon S3, Google Cloud Storage, and Azure Blob Storage provide scalable and durable storage for your image data.



Amazon
S3



These services offer several benefits:

- **Scalability:** Cloud storage can easily scale to accommodate growing image datasets.
- **Durability:** Data is replicated across multiple locations, ensuring high availability and protection against data loss.
- **Cost-effectiveness:** Cloud storage typically follows a pay-as-you-go model, allowing you to pay only for the storage you use.
- **Accessibility:** Image data can be accessed from anywhere with an internet connection, facilitating collaboration and remote processing.

SERVERLESS FUNCTIONS:-

- AWS Lambda;
- Google Cloud Functions;
- Azure Functions.

USE CASES:-

- Image resizing;
- Thumbnail generation;
- Format conversion;
- Other lightweight image processing tasks.

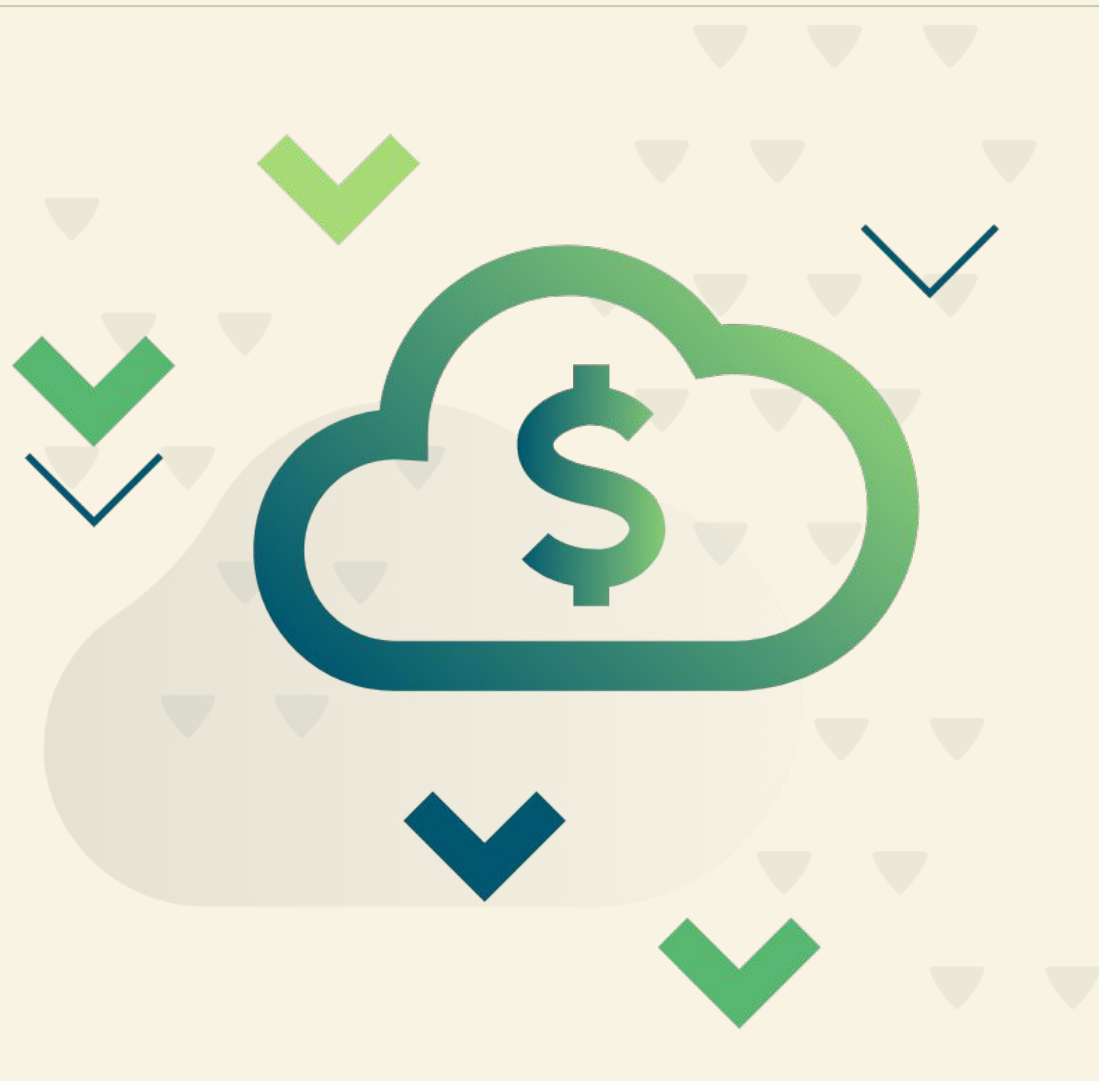
POPULAR CLOUD-BASED IMAGE PROCESSING APIs:

- **Amazon Rekognition;**
- **Google Cloud Vision;**
- **Azure Cognitive Services;**

CAPABILITIES:

- **Object detection;**
- **Image classification;**
- **Facial recognition;**
- **Content moderation;**

...and more!



STRATEGIES FOR OPTIMIZING CLOUD COSTS:-

- Utilize spot instances;
- Right-size resources;
- Leverage reserved instances;
- Monitor and analyze usage patterns;
- Set budget alerts.

Distributed Processing and Parallelism – Data Parallelism

Show how image data can be divided into smaller chunks and processed in parallel across multiple compute nodes.

Data parallelism involves dividing your image data into smaller chunks and processing them in parallel across multiple compute nodes. This approach allows you to leverage the combined processing power of multiple machines, significantly speeding up your pipeline.

Task Parallelism for Concurrent Image Processing

BREAK DOWN IMAGE PROCESSING TASKS
INTO INDEPENDENT SUBTASKS:

Resizing

Filtering

Feature extraction

...and others

APACHE SPARK

- A general-purpose cluster computing framework that excels at handling large datasets and supports various programming languages.

DASK

- A flexible library for parallel computing in Python that provides a familiar interface for working with large arrays and dataframes.

RAY

- A distributed computing framework designed for building and scaling machine learning applications. It offers a simple API for parallelizing Python code and supports various machine learning libraries.

These frameworks provide the tools and abstractions to distribute your image processing workload across multiple machines, enabling you to process large datasets efficiently.

CONTAINERIZATION (E.G., DOCKER):

- allow you to package your applications and dependencies into portable containers that can run consistently across different environments.

ORCHESTRATION (E.G., KUBERNETES):

- automate the deployment, scaling, and management of containerized applications, simplifying the deployment and management of your image processing pipelines.

IDENTIFY PERFORMANCE BOTTLENECKS:

Continuous optimization and cost monitoring are essential to ensure that your **image processing pipelines remain efficient and budget-friendly.**

OPTIMIZE CRITICAL SECTIONS:

Profiling tools help you identify performance bottlenecks in your pipeline by measuring the execution times of different stages. Once you've identified the bottlenecks, you can apply various optimization techniques, such as:

- **Algorithmic improvements:** Explore more efficient algorithms or data structures to speed up critical sections of your code;
- **Hardware acceleration:** Leverage GPUs or specialized hardware accelerators to offload computationally intensive tasks;
- **Parameter tuning:** Experiment with different parameter settings to find the optimal configuration for your workload.

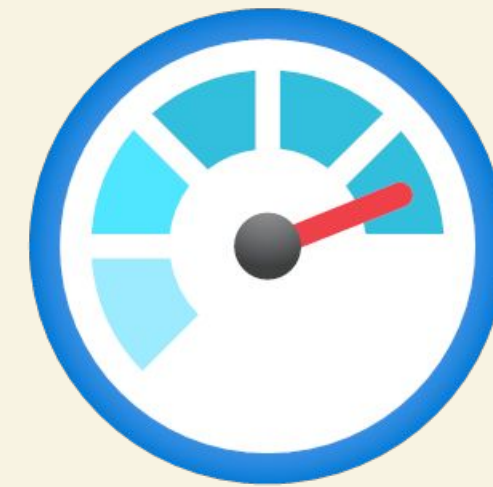


CACHING AND MEMOIZATION FOR SPEEDING UP IMAGE PROCESSING:

- **Caching:**
 - Store the results of computationally expensive operations to avoid recomputing them. This is particularly useful for image transformations or feature extractions that are repeatedly applied to the same images.
- **Memoization:**
 - Cache the results of function calls based on their input arguments. This avoids redundant computations for functions that are called multiple times with the same inputs.

Optimization and Cost Monitoring – Cost Monitoring and Alerting

Cloud providers offer tools and services to track resource usage and associated costs. Setting budget alerts and analyzing usage patterns can help you identify cost optimization opportunities.



Prometheus



DATADOG



DigitalOcean



Amazon Cloudwatch



APPDYNAMICS

new relic

PagerDuty

splunk® >

Additional Insights:

- **The Future of Image Processing:** Advancements in artificial intelligence and ML are poised to revolutionize image processing. Techniques like deep learning and neural networks enable sophisticated image analysis tasks, opening new possibilities for applications in various domains.
- **Ethical Considerations:** As image processing technologies become increasingly powerful, it is crucial to consider their ethical implications. Issues such as privacy, bias, and transparency need to be addressed to ensure responsible and ethical use of image data.



- **Image Data and Its Applications**
- **Challenges Posed by Large Image Datasets**
- **The State of Art**
 - **Choosing the Right Image Formats**
 - **Compression Techniques**
 - **Leveraging Cloud-Native Services**
 - **Distributed Processing and Parallelism**
 - **Containerization and Orchestration**
 - **Optimization and Cost Monitoring**
- **The Future**



Anna Chi

THANK YOU!

📍	31 Harbour Way, London, E14 9TY
☎	738-815-6663
✉	chiheng2013@gmail.com

