

Odessa

Enabling Interactive Perception Applications on Mobile Devices

Grzegorz Milka

10.12.2012

- 1 Problem domain
 - Requirements and characteristics
 - Metrics
 - Methods of adaptation
 - Applications
- 2 Sprout
- 3 Experiment
 - Data Variability Impact
 - Impact of different strategies
- 4 Odessa
 - Design and Implementation
 - Evaluation

Requirements and characteristics

- Crisp-response
- Continuous processing
- Compute-intensive
- Performance depending on data



Metrics

Makespan

Time taken to execute all stages of a data flow for a single thrame.

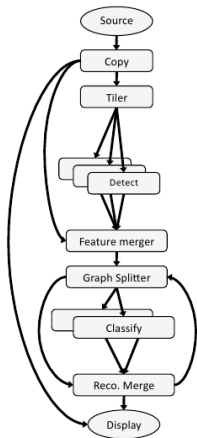
Throughput

Rate at which frames are processed and a measure of the accuracy of the application.

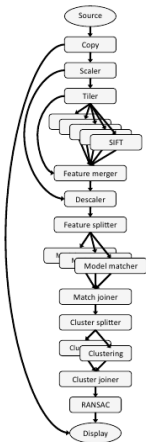
Methods of adaptation

- Offloading
- Pipelining
- Data-parallelism

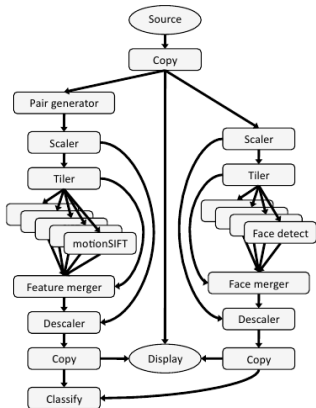
Applications



(a) Face Recognition



(b) Object and Pose Recognition



(c) Gesture Recognition

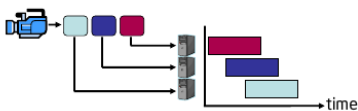
What's sprout

- **Sprout** A distributed stream processing system.
- Uses a data flow model and policy to distribute processing of an application.

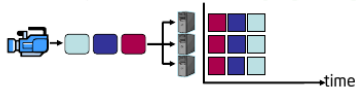
a) Unparallelized vision code: **high latency, low throughput**



b) Inter-frame parallelization: **high latency, high throughput**



c) Intra-frame parallelization: **low latency, high throughput**



Input Data Variability

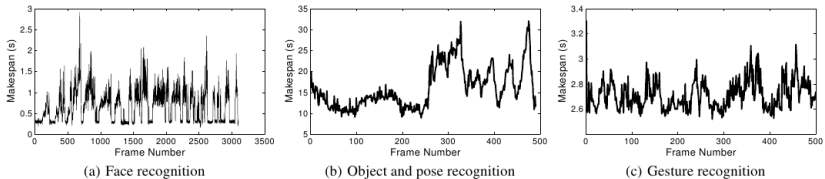


Figure 3: Variation in the per frame makespan.

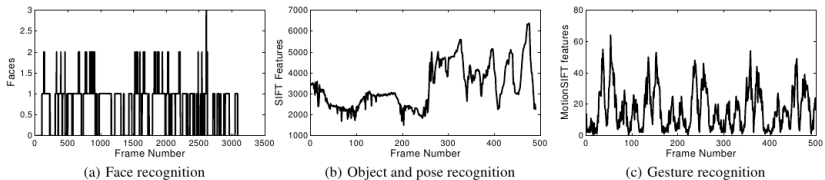


Figure 4: Variation in the number of features extracted per frame.

Variability Across Mobile Platforms

Median speedup

Application	Makespan Laptop	Makespan Netbook	Speedup
Face Recognition	0.078	0.20	2.94
Object and Pose Recognition	1.67	9.17	5.47
Gesture Recognition	0.54	2.34	4.31

Exec time Distribution

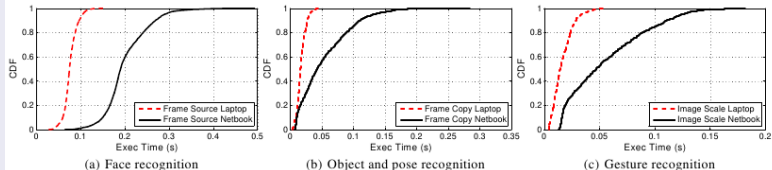
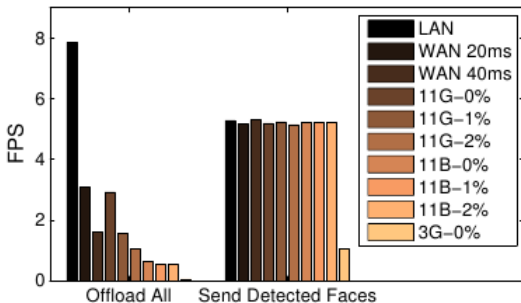


Figure 5: The figures show the variability in the completion time of three example stages running on the laptop and netbook device. These stages perform a fixed operation on the input image that is independent of the scene content.

Network Impact



(b) Frame Rate

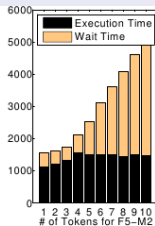
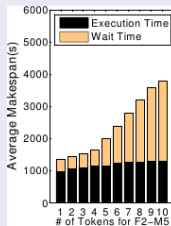
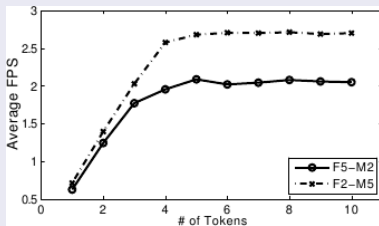
Figure 6: Impact of the network on the performance of the face recognition application.

Effects of Parallelism

Data-Parallelism

# Threads	% Frames with faces	Speedup
1	61.66	1
2	24.87	1.6
3	38.11	2.3

Pipeline Parallelism



Design

3 Goals

- Simultaneously achieve low makespan and high throughput
- Quick response to environment change
- Low computational and communication overhead.

Implementation

Odessa uses **lightweight profiler** which identifies bottleneck. Using this data it employs **greedy** and **incremental** strategy to optimize makespan and throughput.

Decision Engine

```
bottleneck := pick the first entry from the priority heap
if bottleneck is a compute stage then
  a. estimate the cost of offloading the stage
  b. estimate the cost of spawning more workers
else
  if bottleneck is a network stage then
    a. estimate the cost of offloading the source stage
    b. estimate the cost of offloading the destination stage
  end if
end if
take the best choice among a., b. or do-nothing
sleep
```

Odessa's performance and overhead

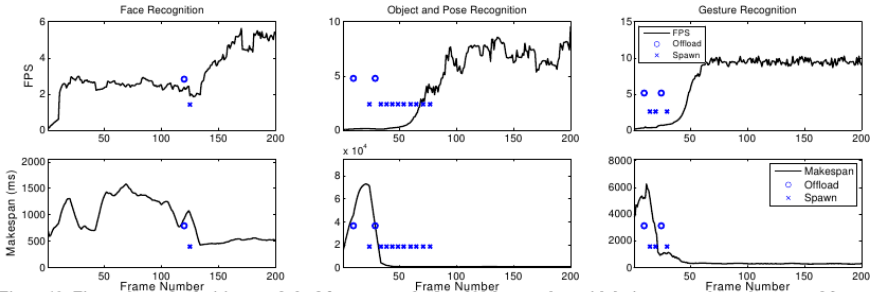
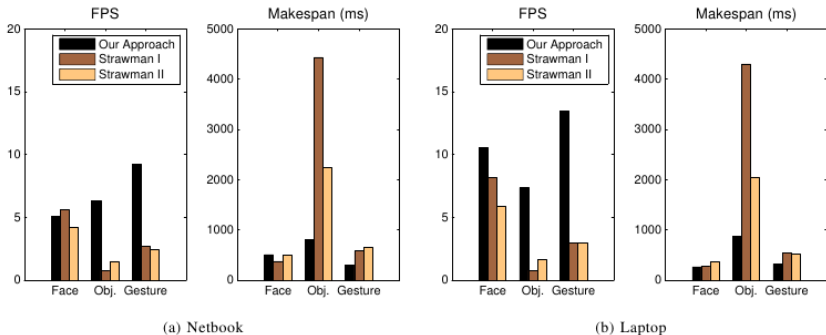


Figure 10: Figure shows the decisions made by *Odessa* across the first 200 frames alongwith the impact on the makespan and frame rate for the three applications on the network.

Odessa's performance and overhead

Application	Netbook	Laptop
Face Reconition	Face detection(2) - 3.39	Nothing - 3.99
Object and Pose Recognition	Object model matching(3), Feature generat- ing(10) - 5.71	Object model matching(3), Feature generat- ing(10) - 5.14
Gesture Recogni- tion	Face detection(1), extracting Motion SIFT features(4) - 3.06	Face detection(1), extracting Motion SIFT features(9) - 5.14

Comparison to other strategies



Comparison to other strategies - Offline Optimizer

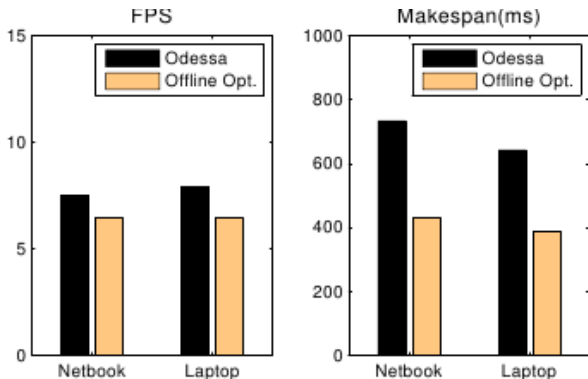


Figure 12: Figures shows the frame rate and makespan achieved by *Odessa* for pose detection, compared to the *Offline-Optimizer*.

Adaptation to Varying Execution Contexts

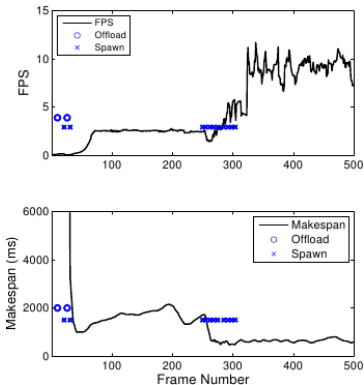


Figure: Core change

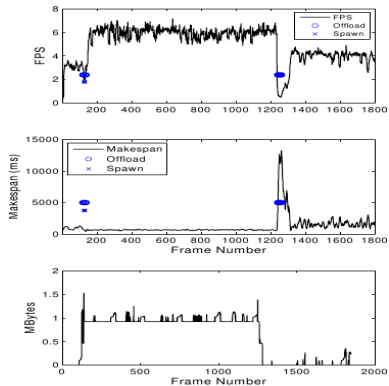




Figure: Network change

Thank you

-  Moo-Ryong Ra, Anmol Sheth, Lily Mummert, Padmanabham Pillai, David Wetherall, Ramesh Govindan
Odessa: Enabling Interactive Perception Applications on Mobile Devices
MobiSys '11 Proceedings of the 9th international conference on Mobile systems, applications, and services
-  Padmanabham Pillai, Lily Mummert, Steven Schlosser, Rahul Sukthankar, Casey Helfrish
SLIPstream: Scalable Low-latency Interactive Perception on Streaming Data
NOSSDAV '09 Proceedings of the 18th international workshop on Network and operating systems support for digital audio and video