CSE 443 OBJECT ORİENTED ANALYSİS AND DESIGN

Homework01 Report

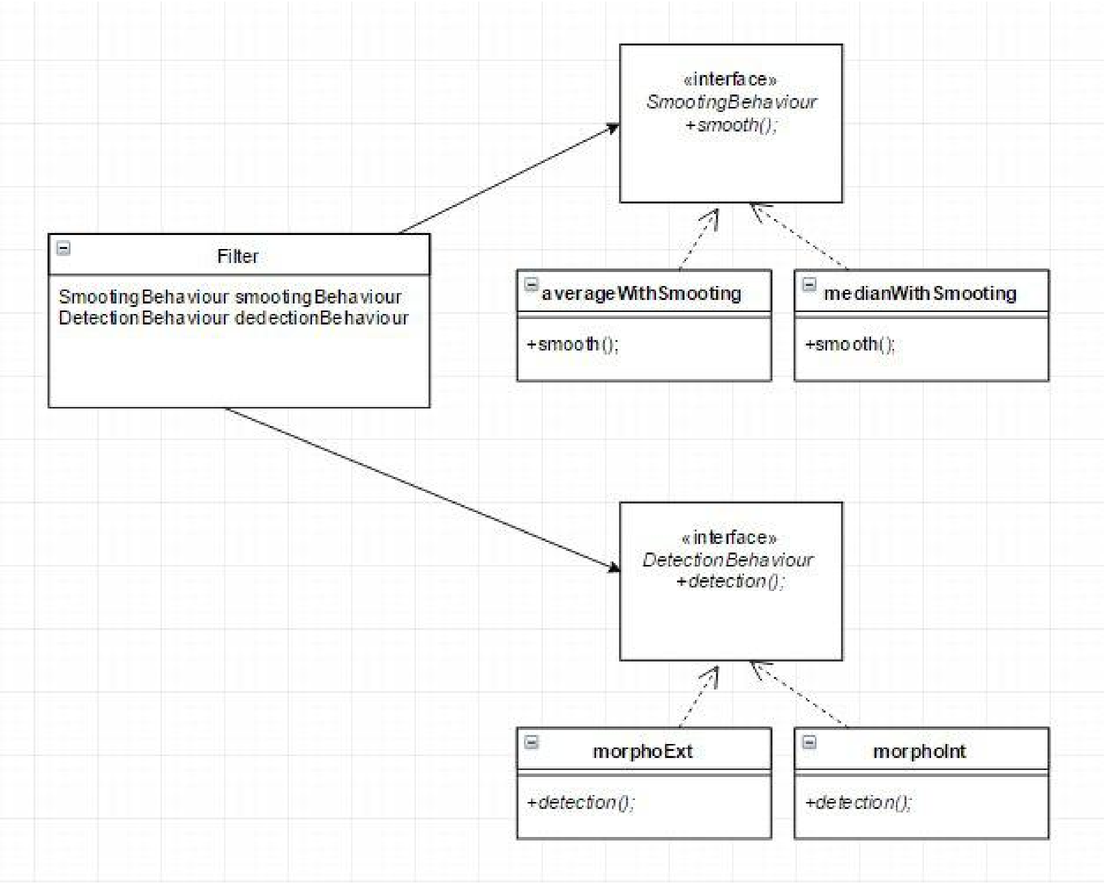
Gebze Technical University
Merve Şahin 121044054

1.) Why do we describe design patterns?

and What are differences design and design pattern ? and then What makes a design solution a “design pattern”?Graphical notations, while important and useful, aren't sufficient. They simply capture the end product of the design process as relationships between classes and objects. To reuse the design, we must also record the decisions, alternatives, and trade-offs that led to it. Concrete examples are important too, because they help you see the design in action.

We describe design patterns using a consistent format. Each pattern is divided into sections according to the following template. The template lends a uniform structure to the information, making design patterns easier to learn, compare, and use.

2.)



3.)Strategy design pattern for solving this problem efficiently and effectively.Here I have the two interfaces,SmootingBehaviour and DedectionBehaviour along with the corresponding classes that implement each concrete behavior.

SmootingBehaviouris an interface that all smooting classes implement.All new smooting classes just need to implement the smooth method

Same thing here for the dedection behaviour; I have an interface that just includes a dedection() method that needs to be implemented

avragewithsmooting : Implementation of smooting for all image that have average

medianWithSmooting: Implementation of smooting for all image that have median

morphoExt: Implementation of dedection for all image that have median and biggest value on mask

morphoInt: Implementation of dedection for all image that have median and smallest value on mask

With this design, other types of objects can reuse our smooth and dedection behaviours because these behaviours are no longer hidden away in my Filter class !!!

And I can add new behaviours without modifying any of our existing behaviour classes or touching any of the FİIter classes that use smooting behaviours

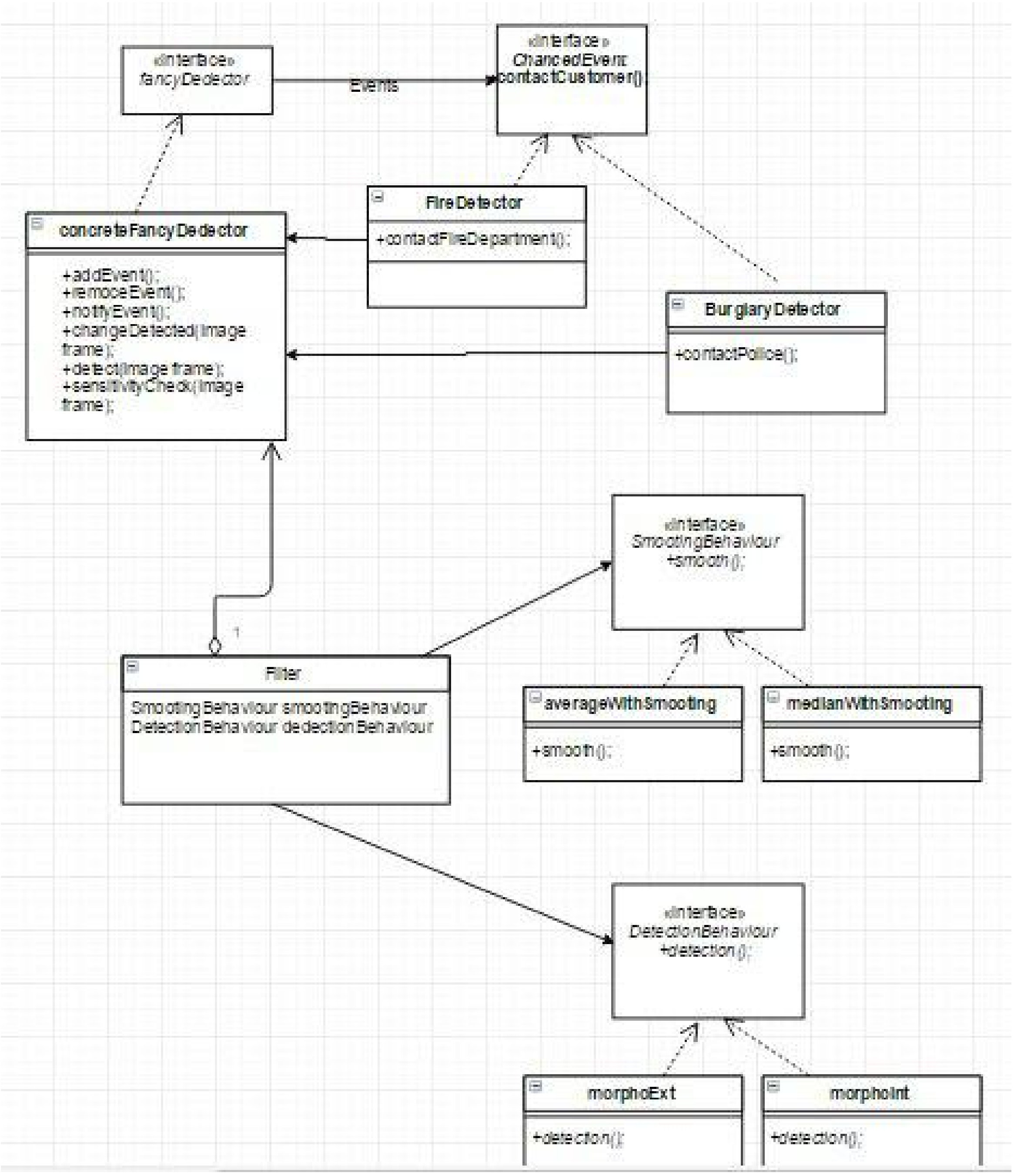
So I encapsulated smooth behavior and dedection behavior and then dedection “algorithms” are interchangeable.

The HAS-A relationship is an interesting one: each filter has a SmootingBehaviour and DedectionBehaviour to which it delegates smooting and dedecting

This is an important technique; in fact,I have been using Strategy design pattern.

4.)Please Can you look at Project file that I can sent to you.

5.)


«interface»
fancyDedector
Events
«interface»
ChancedEvent
contactCustomer();
FireDetector
+contactFireDepartment();
concreteFancyDedector
+addEvent();
+remoceEvent();
+notifyEvent();
+changeDetected(Image frame);
+detect(Image frame);
+sensitivityCheck(Image frame);
BurglaryDetector
+contactPolice();
«interface»
SmootingBehaviour
+smooth();
1
Filter
SmootingBehaviour smootingBehaviour
DetectionBehaviour dedectionBehaviour
averageWithSmooting
+smooth();
medianWithSmooting
+smooth();
«interface»
DetectionBehaviour
+detection();
morphoExt
+detection();
morphoInt
+detection();

I tried to assimilate to observe design pattern above Uml diagram. Teacher!!! I don’t sure The relation between filter class and above structure I didn’t make a desicion arrow type for filter connect to uml diagram