

TUTORIAL ROBOT AND OBJECT DETECTION USING BOOSTED CLASSIFIERS

Dr. Rodrigo Verschae, Advanced Mining Technology Center, Universidad de Chile

Object detection using computer vision methods basically consist on determining the position of a type of object that is present on an image or video. An example of this kind of problem is the detection of robots or the detection of faces. This is a rather difficult problem to solve, mainly because of the variability that the objects can present (different deformations, different poses, presence and absence of structural components, etc.) and because of the variability in the image formation and acquisition process (variable illumination, rotations, translations, different views, scales, etc.) and the variability in the image background. Many methods have been used to solve this problem, however, a type of method that has been able to solve it in the case of robots (as well as other kinds of objects, such as faces and cars) and that can detect multiple object instances in dynamic environments, are those based on statistical learning techniques. Under this paradigm, the object detection problem can be formulated as a two-class classification problem, with the classes being “object” and “non-object”. In particular, using several boosted arranged in a cascade architecture, real-time object detection systems have been obtained.

In this tutorial the current most successfully methods in object detection will be briefly introduced. Especial emphasis will be given to boosting classifiers (e.g. AdaBoost). The process of building an object detection system based on statistical classifiers will be described, as well as the methodologies to used to train the classifiers (e.g. bootstrapping) and the use of a cascade architecture for efficient detection.

If time allows it, the methods currently used to evaluate this kind of system will be presented and an extension of cascade classifiers to the multiclass case will be introduced.

TUTORIAL

DETECCIÓN DE ROBOTS Y OBJETOS USANDO CLASIFICADORES ESTADÍSTICOS DE TIPO BOOSTED

Dr. Rodrigo Verschae, Advanced Mining Technology Center, Universidad de Chile

La detección computacional de objetos consiste básicamente en determinar la posición de un tipo de objetos presentes en una imagen o un vídeo. Un ejemplo de este problema es la detección de robots o la detección de caras. Este es un problema bastante difícil de resolver, principalmente por la variabilidad que pueden presentar los objetos (distintas deformaciones, distintas poses, presencia o ausencia de componentes estructurales, etc.) y por la variabilidad que puede existir en el proceso de adquisición de las imágenes (iluminación variable, rotaciones, traslaciones, distintas vistas y escalas, etc.) y en el fondo de la imagen. Existen muchos métodos que intentan resolver este problema, sin embargo, métodos que han sido capaces de resolverlo en el caso de robots (al igual que en otros objetos, como caras y autos) y que pueden detectar múltiples objetos en ambientes dinámicos, son aquellos basados en técnicas de aprendizaje estadístico. Bajo este paradigma el problema de la detección de objetos se resuelve como un problema de clasificación de dos clases, la clase “objeto” y la clase “no-objeto”. En particular, utilizando métodos de tipo boosted y clasificadores con arquitectura de tipo cascada se han logrado obtener detectores en tiempo real.

En este tutorial se introducirán brevemente los métodos más exitosos de detección de objetos existentes en la actualidad. Se dará especial atención a los clasificadores tipo “boosting” (e.g. AdaBoost) y se describirá en detalle la construcción de un sistema estadístico de detección de objetos, así como las metodologías de entrenamiento a ser utilizadas (boosting y bootstrapping) y una arquitectura de detección eficiente basada en cascadas de clasificadores.

Si el tiempo lo permite, se describirá como evaluar este tipo de sistemas y se presentará una extensión de este tipo de sistemas para el caso de la detección de múltiples tipos de objetos (o múltiples vistas).