Mejora del Reconocimiento de Objetos, Acciones y Efectos Usando los Ofrecimientos Probabilísticos de los Objetos

Por

Esteban Jaramillo Cabrera

Maestro en Ciencias en el área de
Ciencias Computacionales
Instituto Nacional de Astrofísica, Óptica y Electrónica
Noviembre, 2018

Supervisada por:
Dr. Eduardo Morales Manzanares
Dr. José Martínez Carranza

Resumen

El reconocimiento y manipulación de objetos son tareas relevantes para la robótica y la inteligencia artificial en general, teniendo en cuenta los objetos, las acciones y los efectos generados que se han reconocido de manera independiente. Sin embargo, existe una fuerte relación entre éstas tres variables, como lo sugirió Gibson en su teoría de los ofrecimientos[1] (*affordances*). En particular, en esta tesis realizamos el reconocimiento de una interacción a partir de las variables objeto, acción y efecto que se estiman utilizando redes neuronales convolucionales, y luego se capturan sus dependencias probabilistas usando una red bayesiana. Esto es importante, porque al relacionar las variables objeto, acción y efecto podemos mejorar los índices de reconocimiento de cada una de las variables utilizando las relaciones entre ellas. Por otro lado, podemos estimar alguna de las tres variables así no tengamos ninguna información de ésta, siempre y cuando tengamos la información de las otras dos variables, es decir que podemos hacer inferencia con información faltante. Para probar el desempeño del modelo planteado se utilizó la base de datos CERTH-SOR3D de más de 20 mil videos RGB-D involucrando 14 objetos y 13 acciones, a la cual le añadimos información de 7 efectos. Con esta información se construyeron modelos de reconocimiento inicial con una precisión de 71.23% para acciones, 85.02% para los objetos y 82.05% para los efectos. Cuando relacionamos la información de los tres modelos podemos tener este mismo desempeño con menos datos de entrenamiento, pero si usamos los mismos datos para entrenar podemos incrementar el reconocimiento de 71.23% hasta 79.71%, de 85.02% hasta 86.81% para los objetos y pasando de 77.04% a un 82.77% para los efectos. Por otro lado, para labores de inferencia con información faltante se obtuvo una predicción de 63.50%, 24.55% y 76.59% para objetos, acciones y efectos; respectivamente.

[1] El término ***affordances*** del inglés tiene varias traducciones en español, en este texto se utilizará el termino **ofrecimientos** con base en (Jover, 1987)

Abstract

The recognition and manipulation of objects are relevant tasks for robotics and artificial intelligence in general, taking into account the objects, actions and effects generated, which have been recognized independently. However, there is a strong relationship among these three variables, as suggested by Gibson in his theory of affordances. In particular, in this thesis we perform the recognition of an interaction based on the object, action and effect variables that are estimated using convolutional neural networks, and then capture their probabilistic dependencies using a Bayesian network. This is important, because when relating the object, action and effect variables, we can enhance the recognition rates of each of the variables using the relationships between them. On the other hand, we can estimate one of the three variables without having any information about it, as long as we have the information of the other two variables, that is, we can make inference with missing information. To test the performance of the proposed model, we used the CERTH-SOR3D database of more than 20,000 RGB-D videos involving 14 objects and 13 actions, to which we added information on 7 effects. With this information, initial recognition models were constructed with an accuracy of 71.23% for actions, 85.02% for objects and 77.04% for effects. When we relate the information of the three models, we can have this same performance with less training data, but if we use the same data to train, we can increase the recognition of 71.23% up to 79.71% for actions, from 85.02% up to 86.81% for objects and going from 77.04% to 82.77% for the effects. On the other hand, for inference work with missing information, a prediction of 63.50%, 24.55% and 76.59% was obtained for objects, actions and effects; respectively.

Agradecimientos

Dios

Familia

Amigos

PhD Eduardo Morales Manzanares

PhD José Martínez Carranza

Laboratorio de Robótica

Coordinación Ciencias Computacionales

INAOE

CONACYT

Contenido

Lista de Figuras

Lista de Tablas

Lista de Abreviaturas

CNN	Red Neuronal Convolucional *(Convolutional Neural Network)*
BN	Red Bayesiana *(Bayesian Network)*
CPTs	Tablas de Probabilidad Condicional *(Conditional Probabilistic Tables)*
LSTM	Memoria a corto y largo plazo *Long-Short Term Memory*
GST	Espacio-Temporal Generalizado *(Generalized Spatio-Temporal)*
GTM	Coincidencia de Plantillas Generalizada *(Generalized Template-Matching)*
RGB	Rojo, Verde, Azul *(Red, Green, Blue)*
HSV	Matiz, Saturación, Valor *(Hue, Saturation, Value)*
BNT	Caja de herramientas de Redes Bayesianas *(Bayes Net Toolbox)*

SRL	Aprendizaje Relacional Estadístico *(Statistical Relational Learning)*
HCI	Interacción Humano-Computadora *(Human-Computer Interactions)*
ATCRF	Campo Aleatorio Condicional Temporal Anticipatorio *(Anticipatory Temporal Conditional Random Field)*
IR	Incorrectamente Reconocido
CR	Correctamente Reconocido

Capítulo 1

Introducción

Los seres vivos al interactuar con su entorno desarrollan la capacidad de percibir lo que éste les ofrece, para sacar ventaja de todo lo que encuentran a su alrededor. Como lo explica el psicólogo perceptual James J. Gibson en el capítulo 8 “The theory of affordances” de su libro seminal (Gibson, 1978), donde expresa que: “los medios, las sustancias, las superficies, los objetos, los lugares u otros seres tienen ofrecimientos para un ser vivo y éstos pueden verse como ventajas o desventajas”. Por ejemplo, se puede observar que el aire como medio, permite la respiración de algunos seres, pero otros requieren de un medio como el agua para poder realizar la misma acción; también, se puede apreciar cómo el agua ofrece a algunos pequeños insectos una superficie sobre la cual pueden caminar, pero si los animales son de un tamaño y peso considerable requieren de una superficie lo suficientemente rígida y plana para poder llevar a cabo esta acción. Por otro lado, se puede apreciar cómo algunos animales perciben algunos objetos ya sea como juguetes o como herramientas, en el primer caso si se lanza una pelota a un perro éste la percibirá como un juguete al cual perseguirá hasta atraparlo. De modo similar pero más interesante, en el segundo caso, se puede observar cómo algunas especies de primates perciben una piedra como una herramienta para golpear con la cual pueden abrir nueces muy duras. Estos también han aprendido en su entorno que una vara de madera se puede usar como una extensión para extraer su alimento de la madriguera de algún insecto.

Los humanos como animales evolucionados hemos trascendido el concepto de los ofrecimientos, ya que no sólo percibimos y aprovechamos lo que el ambiente nos brinda, sino que adicionalmente hemos modificado el entorno en que vivimos para que éste nos ofrezca mayores ventajas. Por ejemplo, construimos caminos, para tener superficies más estables sobre las cuales podernos desplazar, o casas para tener un lugar en el cual protegernos de un ambiente hostil.

Esta evolución toma mayor relevancia al apreciar la capacidad de los seres humanos para construir y utilizar herramientas, así un objeto puede facilitar alguna tarea en específico. Por ejemplo, un objeto alargado de tamaño y peso moderados permite ser empuñado y se puede usar para golpear, éste sería un palo o martillo; si es usado para arrastrar algo que está más allá de nuestro alcance, sería una especie de rastrillo; además si el objeto cuenta con una punta éste permite perforar. De modo similar en cualquier caso el objeto sería una extensión del brazo que ofrece una ventaja a quien lo utiliza, dependiendo de la tarea que se desee realizar.

1.1 Motivación

La manipulación robótica se desarrolló inicialmente en el campo de la automatización industrial, donde, robots[2] fijos o semifijos como el que se observa en la parte a) de la Figura 1.1 realizan tareas mayormente repetitivas, que requieren un alto nivel de precisión, las cuales pueden llegar a ser desgastantes o peligrosas para los seres humanos. En este caso, se debe tener en cuenta que los entornos suelen ser altamente estructurados, por lo tanto, el robot no necesita de una percepción general, sino que puede determinar su posición en el mundo, al basarse simplemente en sensores propioceptivos combinados con sensores táctiles o finales de carrera.

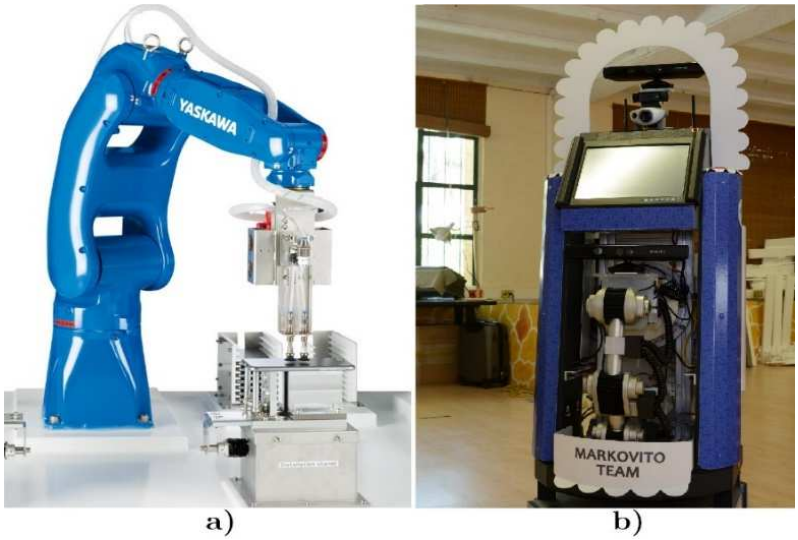


Figura 1.1. a) Manipulador industrial, b) Manipulador de servicio

[2] El término ***robot*** proviene de la palabra checa *robota* que significa "**trabajo forzado**", fue introducida por el dramaturgo y autor checoslovaco Karel Capek, El robot puede ser tanto un mecanismo electromecánico físico como un sistema virtual de software (Zunt, 2002).

La tarea de manipulación cambia notoriamente cuando es realizada por un robot de servicio. Para empezar, usualmente son robots móviles como el que se ve en la parte b) de la Figura 1.1. Éstos se encuentran en ambientes como hogares u oficinas los cuales no son tan estructurados, por lo tanto, se requiere de sensores exteroceptivos para conocer su posición en el mundo. Además, el robot se enfrenta a problemas abiertos, como ordenar un estante o traer un objeto; estas tareas serán normalmente diferentes ya que el robot puede encontrar cantidades y clases de objetos distintos. En este caso se debe tener en cuenta que el ambiente en que se encuentra el robot fue diseñado principalmente para los seres humanos. De aquí se deriva la importancia de que el robot adquiera la capacidad de percibir tanto el entorno como los objetos de forma similar a como lo hacen los seres humanos, basado no sólo en sus características visuales sino también en las "posibles acciones" a realizar con éstos, para así, hacer uso de las herramientas y sacar ventajas de los ofrecimientos presentados por los objetos.

La realización de una tarea por un robot de servicio está compuesta básicamente de 3 pasos.

- Primero se debe percibir el entorno para determinar tanto los elementos presentes como las posiciones en que éstos se encuentran en el mundo y con respecto al robot.
- El segundo paso es la planeación de las acciones a realizar, en este punto se toma la percepción para definir los objetos que intervendrán en la tarea propuesta. Posteriormente, basado en las capacidades del robot y los ofrecimientos presentados por los objetos en la escena, se determina una secuencia de acciones a realizar para alcanzar el estado final del mundo que se desea.
- Por último, el robot ejecuta la secuencia de acciones previamente determinadas en la planeación.

Este proyecto se enfoca principalmente en el primer paso, ya que se realiza un reconocimiento de los objetos, las acciones y los efectos presentes en las interacciones. También de una parte del segundo paso, como lo es la construcción del modelo que permitirá determinar cuáles son los objetos con los que se va a interactuar, las acciones que se van a realizar y los efectos que se esperan obtener.

1.2 Descripción del problema

En robótica la comunidad de investigadores se ha enfocado en diversas tareas dentro de las que se encuentra el reconocimiento de objetos y la realización de tareas de manipulación. Estos enfoques han avanzado de manera importante en los últimos años con el uso de las redes neuronales convolucionales (*CNN, Convolutional Neural Network*) y del aprendizaje por refuerzo. Sin embargo, los reconocimientos de objetos o acciones se hacen de manera independiente y no es común encontrar razonamientos que relacionen las acciones que se pueden hacer con los objetos o los efectos causados.

Los *ofrecimientos de los objetos* se presentan como una relación entre los objetos, las acciones y los efectos. En la literatura éstos se aprendieron inicialmente por exploración como se presenta en (Montesano, et al., 2008) donde un robot realiza un conjunto de acciones, y a partir de un conjunto de objetos obtiene un conjunto de efectos. Las relaciones obtenidas son modeladas en una Red Bayesiana (*BN, Bayesian Network*) que posteriormente se usa para inferir la acción que se debe realizar sobre un objeto para obtener un efecto determinado. Por otro lado, los ofrecimientos también se han aprendido por demostración, donde, un maestro humano realiza un conjunto de acciones sobre un conjunto de objetos y percibe un conjunto de efectos como se presenta en (Koppula & Saxena, 2013). En este caso también se aprenden las relaciones, pero basado en las capacidades de manipulación del maestro humano, ya que el ***agente***[3] no interactúa directamente con los objetos en ningún momento.

En este proyecto la idea principal consiste en relacionar los objetos, acciones y efectos de una interacción entre un humano y un objeto. El conocimiento de dichas relaciones será útil para realizar diversas tareas como las que se observan en la Tabla 1.1. Éstas tareas se dividen en dos tipos, por un lado, están las tareas de inferencia y por el otro las de mejoramiento del reconocimiento. De este modo el principal problema al que nos enfrentamos consiste en establecer un modelo de los ofrecimientos basado en una BN que fusiona la estimación de reconocimiento inicial de acciones objetos y efectos dados por 3 CNNs independientes. Los parámetros de estas redes se aprenden de un paquete de videos RGB-D donde se perciben las interacciones entre un conjunto de maestros humanos y un conjunto de objetos a manipular. Cada una de las interacciones se descompone en 3 partes: los objetos manipulados, las acciones

[3] Un ***agente*** se define como un programa de computadora que tiene cierto grado de autonomía, se comunica con otros agentes y trabaja en beneficio de un usuario en particular (Nwana, 1996)

realizadas y los efectos causados. Adicionalmente se resalta que el uso de una BN como modelo permite lidiar con problemas de información faltante de alguna de las variables.

Tabla 1.1 Tareas de inferencia y mejoramiento de reconocimiento usando los Ofrecimientos

Entrada	**Salida**	**Función**
(A,E)	O	Selección de Objeto
(O,E)	A	Planeación de Acción
(A,O)	E	Predicción de Efecto
(O,A,E)	O	Mejoramiento del Reconocimiento de Objeto
(O,A,E,)	A	Mejoramiento del Reconocimiento de Acción
(A,O,E)	E	Mejoramiento del Reconocimiento de Efecto

Las primeras 3 son tareas de inferencia y consisten en que se tiene información de 2 elementos y se infiere el tercero.

- **Selección de objeto.** Se conoce la acción que se va a realizar y el efecto que se desea obtener, con base en esto se infiere el objeto que se utilizará.
- **Planeación de acción.** Se conoce el objeto que se va a utilizar y el efecto que se desea obtener, con base en esto se infiere la acción que se realizará.
- **Predicción de efecto**. Se conoce el objeto que se va a utilizar y la acción que se desea realizar, con base en esto se infiere el efecto que se obtendrá.

Las otras 3 son tareas de mejora y consisten en que se tiene información de los 3 elementos y se mejora el reconocimiento de cada elemento con la información adicional de los otros 2.

- **Mejoramiento del reconocimiento de objetos**. Se tiene una estimación del objeto utilizado, la acción realizada y el efecto causado, con base en esto se infiere el objeto utilizado con mayor probabilidad de acierto.
- **Mejoramiento del reconocimiento de acciones**. Se tiene una estimación de la acción realizada, el objeto utilizado y el efecto causado, con base en esto se infiere la acción realizada con mayor probabilidad de acierto.

- **Mejoramiento del reconocimiento de efectos**. Se tiene una estimación del efecto causado, el objeto utilizado y la acción realizada, con base en esto se infiere el efecto causado con mayor probabilidad de acierto.

1.3 Objetivos

1.3.1 Objetivo general

Establecer una metodología que, a partir del modelado de los ofrecimientos en una red bayesiana, permita mejorar el reconocimiento de los objetos, las acciones y los efectos que componen cada interacción, lidiando con la incertidumbre y permitiendo realizar tareas de inferencia como selección de objetos, planeación de acciones o predicción de efectos.

1.3.2 Objetivos específicos

- Obtener a partir de cada video de la base de datos, información discriminante que permita estimar de forma independiente los objetos, las acciones y los efectos presentes en una interacción.
- Especificar la arquitectura y parámetros para entrenar la CNN para estimar el reconocimiento inicial de los objetos.
- Especificar la arquitectura y parámetros para entrenar la CNN para estimar el reconocimiento inicial de las acciones.
- Especificar la arquitectura y parámetros para entrenar la CNN para estimar el reconocimiento inicial de los efectos.
- Establecer el modelo de la BN que codifica los ofrecimientos que relacionan los objetos, las acciones y los efectos.
- Usar el modelo planteado en tareas de mejoramiento de reconocimiento e inferencia.

1.4 Descripción de la metodología para modelar los ofrecimientos de los objetos

Para el desarrollo de este proyecto se plantea una metodología que consta de tres fases, como se observa en la Figura 1.2. Para empezar, se tiene como entrada un video RGB-D. El cual pasa inicialmente por una fase de preprocesamiento, en la cual se computa, tanto la segmentación como el flujo óptico de los videos. Después se

calcula un reconocimiento inicial, basado en tres CNNs independientes entre sí, que hacen una estimación inicial de las 3 componentes de la interacción, objetos, acciones y efectos. Posteriormente se tiene una fase de fusión; en ésta la idea es utilizar la información extra de los ofrecimientos que se codifican en una BN, donde se combinan las estimaciones de la etapa previa para mejorar los reconocimientos. Adicionalmente, el modelo obtenido es útil en otras tareas de inferencia como la selección de objetos, planeación de acciones o predicción de efectos.

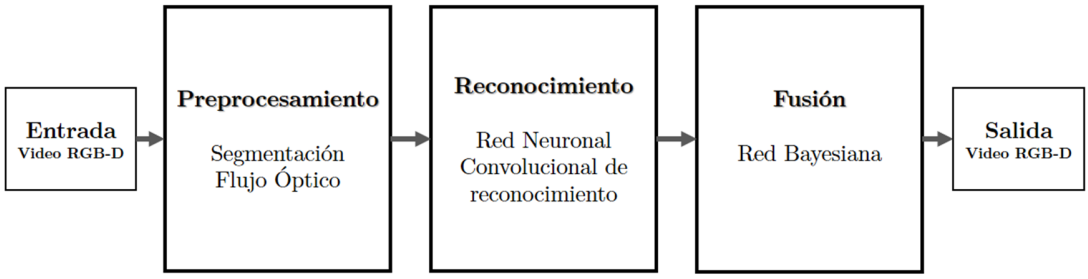


Figura 1.2 Estructura general del modelo de ofrecimiento

1.5 Contribuciones

Las tareas de reconocimiento de objetos y acciones son de alto interés y además han alcanzado en los últimos años porcentajes de acierto comparables con las capacidades humanas. Esto gracias al uso de las CNNs que ofrecen un alto poder decisivo, pero el poder descriptivo queda relegado a un segundo plano. Para el desarrollo de este proyecto se plantea combinar la capacidad de decisión de las CNNs en el reconocimiento de objetos, acciones y efectos, con el poder de descripción de una BN que modela el concepto de "Ofrecimientos de los objetos" ya que codifica las relaciones entre los tres elementos mencionados, de este modo se enriquece la percepción de las interacciones y es útil en diversas tareas.

Otro punto a resaltar es que la base de datos con la que se trabaja cuenta inicialmente con un conjunto de 54 interacciones que relacionan 14 objetos con 13 acciones. Pero en este proyecto se han adicionado 7 efectos a las relaciones ya existentes. Esto entrega mayor información que es de gran utilidad tanto en las tareas de mejoramiento del reconocimiento como en las tareas de inferencia.

También es relevante el hecho de que en los trabajos relacionados se han enfocado en resolver problemas específicos, planteando una red diferente en cada caso.

Mientras que la red presentada en este proyecto exhibe propiedades multiobjetivo ya que la BN se estructura y aprende una sola vez y puede ser usada en todas las tareas de inferencia y reconocimiento mencionadas anteriormente.

1.6 Limitaciones

Este proyecto no planea alcanzar los valores del trabajo relacionado en las tareas de reconocimiento de objetos o acciones, ya que se trabaja con CNNs relativamente básicas para el reconocimiento inicial. Lo importante aquí es demostrar cómo la información adicional al relacionar los objetos con las acciones y los efectos por medio de los ofrecimientos modelados en un BN, mejora el reconocimiento inicial que se tiene y además se pueden realizar tereas de inferencia.

Otro punto a tener en cuenta es que el modelo planteado se usará en tareas de inferencia, pero no se ejecutará una manipulación por parte de un robot. Esto presenta una limitante a la hora de evaluar la inferencia ya que en la literatura se evalúa basado en el éxito de la ejecución de la interacción, pero en nuestro caso el éxito solo compara la estimación con la etiqueta previamente definida.

1.7 Organización de la tesis

El resto de la tesis se estructura de la siguiente forma. En el capítulo 2 se encuentra el marco teórico, donde se presenta el concepto de los ofrecimientos y los modos de aprendizaje que se han usado, además de una presentación de los conceptos de red neuronal convolucional, red bayesiana, segmentación de una imagen y flujo óptico. En el capítulo 3 se encuentran los trabajos relacionados, donde inicialmente se presentan 4 equipos de trabajos que han desarrollado investigaciones alrededor de los ofrecimientos, presentando diferentes modelos, utilizándolos en diferentes tareas. En el capítulo 4 está la explicación del método propuesto basado en tres fases conectadas y se explica cómo trabaja cada una de ellas. En el capítulo 5 se consignan los experimentos y resultados obtenidos, partiendo del establecimiento de los diferentes parámetros usados tanto en las redes neuronales convolucionales como en la red bayesiana. Por último, en el capítulo 6 están las conclusiones y el trabajo futuro.

Capítulo 2

Marco teórico

En este capítulo se presentan los conceptos principales utilizados en el desarrollo del proyecto, empezando por los ofrecimientos, teniendo en cuenta cómo se han modelado en la literatura y las diferentes tareas que se han desempeñado con ellos. Posteriormente se presenta el concepto de red bayesiana, el cual será la base que permitirá modelar los ofrecimientos. También se presenta una descripción básica de las redes neuronales convolucionales, ya que éstas constituyen la etapa de entrada al modelo principal. Por último, se presenta la etapa de procesamiento de imágenes y videos, ya que se obtienen tanto segmentación de imágenes como flujos ópticos, para representar la información útil para el reconocimiento de las componentes (objetos, acciones y efectos) de cada interacción en la base de datos.

2.1 Los ofrecimientos de los objetos

Los ofrecimientos de los objetos fueron presentados inicialmente en el campo de la psicología perceptual por James J. Gibson (Gibson, 1978) como una relación (Objeto-Acción) la cual se denominó "posibles acciones"; es decir, que un objeto ofrece un conjunto de acciones que un sujeto puede llevar a cabo con él. Posteriormente este concepto se importó al campo de las ciencias computacionales en (Montesano, et al., 2007), donde dicha relación fue extendida a (Objeto-Acción-Efecto) como se observa en la Figura 2.1. Estas relaciones se pueden utilizar en diferentes tareas como lo vimos en la Tabla 1.1. Se debe tener en cuenta que dichas relaciones, que representan conocimiento, se han aprendido ya sea por experimentación o por demostración dependiendo de la tarea que se desee realizar.

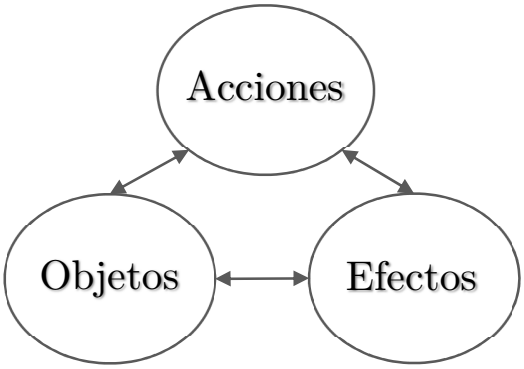


Figura 2.1 Relación entre Objetos-Acciones-Efectos en los ofrecimientos.

2.1.1 Aprendizaje por experimentación

Para el aprendizaje de los ofrecimientos por experimentación, se parte de la idea de que se tiene un robot, el cual cuenta con la capacidad de realizar acciones básicas sobre objetos y adicionalmente tiene la capacidad de percibir tanto los objetos como los efectos generados sobre éstos al realizar las diferentes acciones. Luego se le permite al robot experimentar con el entorno al realizar las acciones básicas sobre los objetos mientras se perciben los efectos, así el robot va generando un mapa de conocimiento del mundo ya que va aprendiendo cómo afecta al mundo (efectos generados), cuando realiza una acción con un objeto determinado.

Este esquema de aprendizaje fue presentado en (Montesano, et al., 2007) donde se dota el robot con un conjunto de habilidades motoras {agarrar, tocar, golpear} y habilidades de percepción {forma del objeto, tamaño del objeto, velocidad del objeto, velocidad de la mano, distancia objeto-mano, contacto}, con las cuales se inicia una etapa denominada *balbuceo*, en la cual el agente intenta realizar las acciones previamente definidas con los objetos y percibe cuáles son los efectos producidos, como se observa en la Figura 2.2 donde se aprecia inicialmente un robot frente a un conjunto de objetos (cubos y esferas de diferentes colores y tamaños) y luego se observa cómo el agente intenta realizar las acciones de agarrar (parte superior de la imagen) y golpear (parte inferior de la imagen), mientras percibe el objeto determinado, en este caso las relaciones entre las variables se modelan en una BN. Este tipo de aprendizaje es útil en tareas que posteriormente implican una interacción entre el robot y los objetos ya que al haber pasado por una etapa de experimentación el agente tiene presente sus ventajas y limitaciones al interactuar con su entorno.

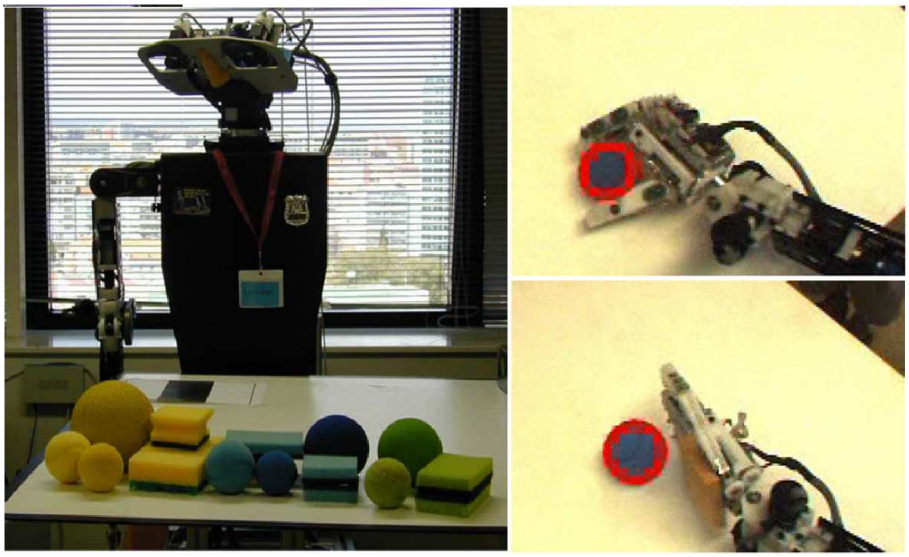


Figura 2.2 Ejemplo de aprendizaje de los ofrecimientos por experimentación (Montesano, et al., 2007), a la izquierda está el robot y los objetos con los que la interactuar, a la derecha está cómo percibe el robot los objetos y las acciones.

2.1.2 Aprendizaje por demostración

En el aprendizaje de los ofrecimientos por demostración, se parte de la idea de que se tiene un agente, el cual tiene la capacidad de percibir tanto los objetos como los efectos generados con éstos al realizar las diferentes acciones, pero no requiere la capacidad de ejecutar dichas acciones, ya que éstas serán realizadas por un maestro ya sea un ser humano o un robot diferente al agente, mientras el agente se limita a percibir las diferentes interacciones y de este modo va generando un mapa de conocimiento.

Este esquema de aprendizaje fue presentado en (Koppula, 2015), donde al agente se le suministra un conjunto de interacciones, grabadas en videos, en los cuales un grupo de maestros humanos interactúa con diferentes objetos. De este modo, el agente establece una base de conocimiento en la cual relaciona los objetos con las acciones y los efectos. Esto se observa en la Figura 2.3, donde inicialmente se ve una persona que va a interactuar con una taza, a partir de esta información el agente es capaz de anticipar las posibles acciones que se van a realizar y se calculan las posibles trayectorias a seguir, las cuales se representan con un mapa de calor en la segunda parte de la imagen. Este tipo de aprendizaje de los ofrecimientos es útil en tareas de reconocimiento, ya que las relaciones aprendidas entre objetos, acciones y efectos se pueden tomar como información adicional que enriquece la percepción.

Figura 2.3 Ejemplo de aprendizaje de los ofrecimientos por demostración (Koppula, 2015), a la izquierda está un fotograma de un persona agarrando un pocillo, a la derecha están las trayectorias de las posibles acciones que se van a realizar con el objeto.

Para nuestro proyecto de tesis, los ofrecimientos de los objetos se aprenderán por demostración a partir de un conjunto de maestros humanos grabados en videos RGB-D. El modelo seleccionado para representar la base de conocimiento que relaciona (Objeto-Acción-Efecto) será una red bayesiana, cuyo concepto se presenta a continuación de forma general.

2.2 Red Bayesiana (BN)

Una BN es un modelo gráfico dirigido que representa la distribución de probabilidad conjunta de un conjunto de variables aleatorias X = {X1, ..., Xn} (Sucar, 2015). Estos grafos están compuestos de nodos y arcos dirigidos que los conectan entre sí, los nodos representan las variables aleatorias y los arcos las dependencias directas entre las variables X, de este modo si se tiene un arco que va de Xi a Xj, se dice que Xi es un padre de Xj, así la distribución de probabilidad de Xj depende de los valores de Xi. Entonces la estructura del grafo determinado, codifica un conjunto de relaciones condicionales de dependencia entre las diferentes variables. Las BN = (G, Θ) se definen a partir de la estructura de su grafo G y un conjunto de parámetros locales $\Theta = \{\theta i\}$, los cuales son las probabilidades condicionales para cada variable dados sus padres en el grafo. Una vez definida una BN a partir de su estructura y parámetros, se pueden realizar tereas de inferencias como se explica más adelante. Estas responden consultas probabilísticas conocidas como (Xi | padres (Xi), θi) de cada nodo en el grafo dependiendo de los padres de (Xi).

2.2.1 Representación e inferencia de la BN

La representación de una BN está dada a partir de su estructura y parámetros asociados, por otro lado, la inferencia está relacionada con las deducciones que se pueden llevar a cabo con dicha red.

- **Estructura**

La estructura corresponde al grafo G que determina las relaciones de independencia condicional de la distribución de probabilidad conjunta. Ésta puede ser definida por un experto o puede ser aprendida directamente de los datos, otra alternativa es combinar conocimiento subjetivo del experto con aprendizaje. Para ello se parte de la estructura dada por el experto, la cual se valida y mejora utilizando datos estadísticos.

En cuanto a las técnicas de aprendizaje estructural, éstas dependen del tipo de estructura de red ya sea un árbol, poli-árbol o red multiconectada. Para el aprendizaje de estructuras de árboles se tiene el algoritmo desarrollado por (Chow & Liu, 1968), para aproximar una distribución de probabilidad por un producto de probabilidades de segundo orden. Luego (Rebane & Pearl, 1987) extendieron el algoritmo de Chow y Liu para poli-árboles basándose en la distinción entre los tres posibles tipos de grupos de tres nodos adyacentes permitidos en un gráfico acíclico dirigido (DAG). Parten del esqueleto obtenido con Chow y Liu para luego determinar las direcciones de los arcos utilizando pruebas de dependencia entre tripletas de variables. Por último, para las redes multiconectadas existen dos clases de métodos para el aprendizaje genérico de redes bayesianas. Por un lado, están los métodos basados en medidas de ajuste y búsqueda (Friedman, et al., 1997), (Friedman, et al., 2000). Por el otro lado están los métodos basados en pruebas de independencia (Pearl, 2002) (Spirtes & Glymour, 2016).

- **Parámetros**

Para especificar completamente una BN y así representar la distribución de probabilidad conjunta, es necesario especificar para cada nodo X su distribución de probabilidad condicional dados sus padres en el grafo.

Teniendo en cuenta que los parámetros son relativos al tipo de nodo y suponiendo una distribución discreta se tiene que:

- Nodos de raíz: se especifican por un vector de probabilidades marginales.

• Otros nodos: se especifican por las tablas de probabilidad condicional (CPT) de la variable dados sus padres en el grafo.

En este punto se debe tener en cuenta que, para el caso de variables discretas, el número de parámetros en un CPT aumenta exponencialmente con el número de padres de un nodo. Esto puede volverse problemático cuando hay muchos padres, los requisitos de memoria pueden llegar a ser muy grandes, y también es difícil estimar tantos parámetros. Se han propuesto dos alternativas principales para superar este problema (Sucar, 2015), la primera se basa en modelos canónicos y la segunda en representaciones gráficas de CPTs.

- **Inferencia**

La inferencia probabilística consiste en propagar los efectos de cierta evidencia en una red bayesiana (Sucar, 2015), para estimar su efecto sobre las variables desconocidas; es decir, al conocer los valores para algún subconjunto de variables en el modelo, se obtienen las probabilidades posteriores de las otras variables. En el caso particular de que el subconjunto de variables desconocidas este vacío; se obtienen las probabilidades previas de todas las variables.

Básicamente hay dos variantes del problema de inferencia en redes bayesianas (Sucar, 2015). Uno es obtener la probabilidad posterior de una sola variable, H, dado un subconjunto de variables conocidas **E**, es decir, P (H | **E**). Específicamente, estamos interesados en las probabilidades posteriores de las variables desconocidas en el modelo, esta es la aplicación más común, y lo denominaremos inferencia de consulta única. La segunda variante consiste en calcular la probabilidad posterior de un conjunto de variables, **H** dada la evidencia, **E**, es decir, P (**H** | **E**). Esto se conoce como consulta de inferencia conjunta. En principio, se puede resolver usando la inferencia de consulta única varias veces aplicando la regla de la cadena, lo que lo convierte en un problema un poco más complejo que el caso anterior.

Como se ha mencionado anteriormente, para este proyecto el modelado de los ofrecimientos está dado por una BN cuya estructura fue definida subjetivamente y los parámetros de los CPTs son calculados a partir de los datos. Esta red que recibe como variables conocidas de entrada, los vectores de las estimaciones de las acciones, los objetos y los efectos, éstos son un conjunto de probabilidades marginales dadas por un grupo de CNNs, cuyo concepto básico se presentará a continuación.

2.3 Red Neuronal Convolucional (CNN)

Las CNNs están basadas en el Neocognitron, de (Fukushima, 1980), donde se presenta una red auto-organizada que aprende sin necesidad de un maestro y adquiere la capacidad de reconocer estímulos de patrones visuales basados en la similitudes geométricas de forma, sin verse afectada por la posición en que se encuentre dentro de la imagen. Posteriormente el proceso fue mejorado por (LeCun, et al., 1998) quien introdujo en el entrenamiento, el método de aprendizaje basado en retro-propagación (*Backpropagation*), demostrando que dada una arquitectura apropiada de red se pueden usar algoritmos de aprendizaje basados en gradiente para sintetizar una superficie de decisión compleja que puede clasificar patrones de alta dimensión. Recientemente las CNNs fueron refinadas en (Ciresan, et al., 2012), quien las implementó en un GPU consiguiendo un proceso de entrenamiento más rápido de lo que se tenía anteriormente, aumentando así el uso de las mismas. Dichas redes se componen de neuronas que tienen pesos y sesgos que pueden aprender. Cada neurona recibe algunas entradas, realiza un producto escalar y luego aplica una función de activación. Al igual que el perceptrón multicapa, también se tendrá al final de la red una capa totalmente conectada que representa una función de pérdida o costo tipo Softmax.

2.3.1 Arquitectura de la CNN

En general las CNNs se utilizan como redes de clasificación. Están construidas con una estructura que presenta inicialmente una fase de extracción de características, compuesta de neuronas convolucionales y de reducción de muestreo. Para finalizar con neuronas de tipo perceptrón para realizar la clasificación final sobre las características extraídas.

- **Capa convolucional**

Este tipo de capas son las que componen la fase de extracción de características de la red al utilizar una operación llamada convolución, en lugar de utilizar la multiplicación de matrices que se aplica en el perceptrón, por lo tanto, éstas le dan el nombre a la red.

En esta operación de convolución, que se observa en la Figura 2.4, se recibe como entrada la imagen o un mapa de características calculado en una capa anterior, luego se toma una ventana deslizante y se realiza una combinación lineal de los pixeles de

entrada con cada filtro, generando así cada pixel de salida del mapa de características de la imagen original, aquí se debe tener en cuenta que para n filtros, se genera un mapa de características de profundidad n.

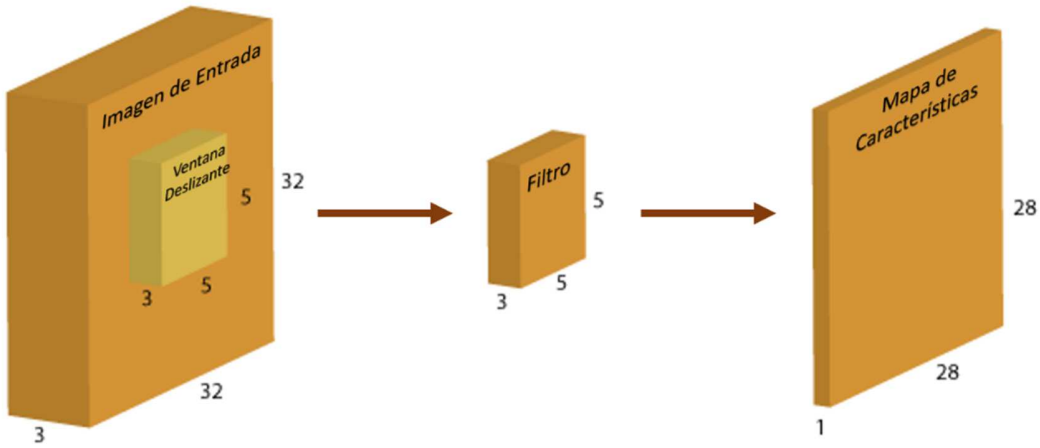


Figura 2.4 Ejemplo de una capa convolucional, donde se observa una imagen de entrada que se convoluciona con un filtro para obtener un mapa de características.

La ventaja presentada por esta red es que un mismo filtro sirve para extraer una misma característica en cualquier parte de la matriz de entrada, con esto se consigue reducir el número de conexiones y el número de parámetros a entrenar en comparación con una red multicapa totalmente conectada.

- **Capa de reducción o de *pooling***

La capa de reducción se ubica generalmente después de la capa convolucional. Esta ayuda a disminuir la cantidad de parámetros, al quedarse con las características más comunes ya que reduce las dimensiones espaciales de ancho y alto del volumen de entrada para la siguiente capa convolucional, teniendo en cuenta que ésta no afecta a la dimensión de profundidad.

Originalmente en las CNNs utilizaban un proceso de sub-muestreo para llevar a cabo esta operación. Sin embargo, estudios recientes han demostrado que otras operaciones, como por ejemplo *max-pooling* que se explican párrafos abajo, son mucho más eficaces en resumir características sobre una región. Sin embargo, una reducción de este tipo puede ser beneficioso para la red por dos razones. Primero, la disminución en el tamaño conduce a una menor sobrecarga de cálculo para las próximas capas de la red; segundo, también ayuda a reducir el sobreajuste.

La operación *max-pooling*, como se observa en la Figura 2.5, donde se divide la imagen de entrada en un conjunto de rectángulos y se queda con el valor máximo de cada uno de ellos. Así se obtiene como resultado que el tamaño de los datos se reduce por un factor igual al tamaño de la ventana de muestra sobre la cual se opera.

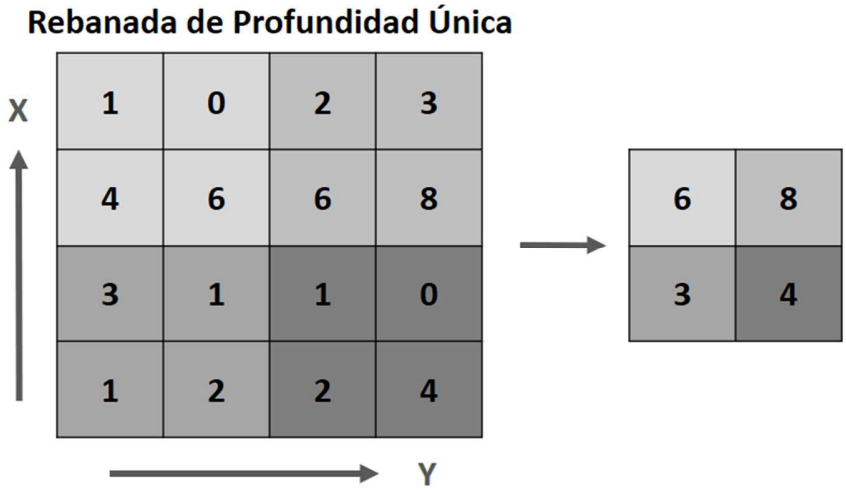


Figura 2.5 Ejemplo de capa de reducción (2×2) cada valor de la salida, es el valor mayor de una ventana (2×2) de la entrada

- **Capa clasificadora totalmente conectada**

Después de la fase de extracción de características dada por las capas convolucionales y de reducción, los datos finalmente llegan a la fase de clasificación. Las neuronas en esta etapa funcionan de manera idéntica a las de un perceptrón multicapa, como se observa en la Figura 2.6, donde cada neurona de una capa está conectada a todas las neuronas de la capa anterior, de esta manera se llega a la última capa clasificadora, la cual tendrá tantas neuronas como el número de clases que se desea predecir.

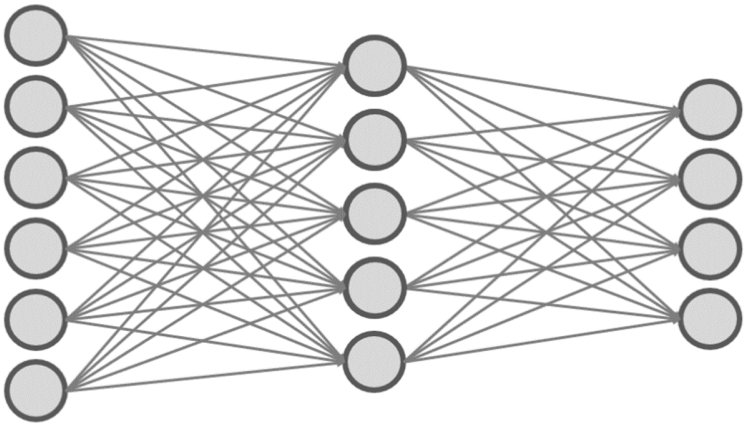


Figura 2.6 Ejemplo de capa totalmente conectada

- **Función *SoftMax***

Como se mencionó anteriormente, en la última capa de la red se tienen K neuronas, una para cada clase que se desea predecir, así, la respuesta de la red en este punto es un vector de K números reales. Dicho vector se convierte a un conjunto de valores entre 0 y 1 cuya suma es igual a 1, este vector representa la probabilidad de reconocimiento y se obtiene por medio de la función softmax, la cual es empleada en varios métodos de clasificación multiclase, tales como Regresión Logística Multinomial, clasificadores Bayesianos ingenuos (*naive Bayes*) o Redes Neuronales Artificiales.

En matemáticas, la función *softmax*, o función exponencial normalizada; es una generalización de la función logística. Se emplea para "comprimir" un vector K-dimensional, z, de valores reales arbitrarios en un vector K-dimensional, $\sigma(z)$, de valores reales en el rango [0, 1]. La función está dada por la expresión (2).

$$\sigma: \mathbb{R}^K \rightarrow [0,1]^K \tag{1}$$

$$\sigma(\boldsymbol{z})_j = \frac{e^{z_j}}{\sum_{k=1}^{K} e^{z_k}} \qquad para \quad j = 1, \dots, K \tag{2}$$

2.3.2 Entrenamiento de la CNN

Una vez que se tiene una arquitectura definida, se pasa a la etapa de entrenamiento, la cual consiste en el aprendizaje de los pesos de la red que permiten calcular los diferentes mapas de características hasta llegar a la capa final donde se realiza el reconocimiento. En este punto es importante tener en cuenta que el entrenamiento requiere un conjunto grande de datos de entrada, compuesto de imágenes y etiquetas. Dicho conjunto debe ser grande ya que se requieren calcular una gran cantidad de pesos de la red y si no se cuenta con suficiente información de entrada se puede caer en sobreajustes.

Al pasar al entrenamiento propiamente dicho se siguen un conjunto de pasos (LeCun, et al., 1998). Primero, se inicializan todos los parámetros o pesos de la red con valores aleatorios. Segundo, se toma una imagen de entrada la cual se propaga a través de la red calculando los diferentes mapas de características, obteniendo una estimación final. Tercero, se calcula el error total de las probabilidades resultantes del modelo al comparar la estimación obtenida con la etiqueta de la entrada seleccionada. Cuarto, se propaga hacia atrás para calcular el error de gradiente de todos los pesos

en la red, utilizando el gradiente descendiente para actualizar estos valores y minimizar el error de salida. Los pasos de dos al cuatro se repiten un número determinado de veces con diferentes imágenes, pero teniendo en cuenta que el entrenamiento de la red no se realiza imagen por imagen, sino que se tienen grupos de éstas, conocidos como lotes.

En el desarrollo de esta tesis se decidió utilizar CNNs para establecer el reconocimiento inicial de objetos, acciones y efectos, ya que estas ofrecen buenas características para los reconocimientos basados en imágenes. Para estas redes se seleccionó una arquitectura de red compuesta de 5 capas, las primeras dos son convolucionales, la tercera y cuarta son totalmente conectadas y por último se tiene una capa de softmax, la cual entrega la estimación en los diferentes casos, esto se puede apreciar con mayor detalle en la sección 4.3. También se definieron un conjunto de parámetros de entrenamiento como tamaño del lote igual a 64 imágenes y 20000 pasos de entrenamiento para cada red. De igual forma esto se puede apreciar con mayor detalle en la sección 5 donde se presentan y analizan los resultados obtenidos.

2.4 Procesamiento de imágenes y videos

El procesamiento de imágenes digitales consiste en la aplicación de algoritmos sobre los valores de los pixeles de una imagen de entrada que puede ir desde una imagen en escala de grises (1D) pasando por RGB (3D) y RGB-D (4D) llegando a imágenes médicas o hiper-espectrales (>4D), para obtener ya sea una nueva imagen o una modificación de la misma de entrada o para obtener un conjunto de características asociadas a dicha entrada. Hoy en día, el procesamiento de imágenes se encuentra entre los campos con mayor crecimiento dentro de la ciencia en general, de este modo el uso de las imágenes o los videos para llevar a cabo diferentes tareas se ha vuelto común.

El propósito del procesamiento de imágenes se divide en 5 grupos. Son:

- Visualización: para observar los objetos que no son visibles.
- Restauración de la imagen: para crear una mejor imagen.
- Recuperación de imágenes: para buscar una imagen de interés.
- Medición del patrón: para medir varios objetos en una imagen.
- Reconocimiento de imágenes: para distinguir los objetos en una imagen.

En el desarrollo de este proyecto se trabajará en el campo del reconocimiento de imágenes, pero inicialmente se lleva a cabo un proceso de segmentación, con el objetivo de eliminar información de poca relevancia para el reconocimiento de los objetos. También se trabajará con videos, sobre los cuales se calcula el flujo óptico el cual permite obtener información de un intervalo de tiempo para llevar a cabo el reconocimiento tanto de las acciones como de los efectos, presentes en las diferentes interacciones.

2.4.1 Segmentación de imágenes

El concepto de segmentación parte de la premisa de que en general la mayoría de las imágenes están constituidas por regiones o grupos de pixeles que tienen características homogéneas como color, nivel de gris o textura, entre otros. La segmentación es el proceso en el cual se asigna una determinada etiqueta para cada elemento de la imagen, dicha etiqueta cataloga cada pixel en un grupo determinado con características similares.

La segmentación se puede llevar a cabo por varios métodos diferentes, éstos se pueden agrupar en 2 grandes familias (Linda & Shapiro, 2001). Por un lado, se tienen los que aprovechan las discontinuidades o cambios grandes del nivel de gris entre los pixeles, algunas técnicas que utilizan este criterio son la detección de líneas, bordes o puntos aislados. Por otro lado, están los que aprovechan las similitudes de niveles de gris, así, las divisiones de la imagen se hacen agrupando los pixeles que tienen unas características similares, algunas técnicas que usan esto son la umbralización o el crecimiento de regiones, entre otras. La selección de la técnica adecuada de segmentación depende del objetivo buscado.

Un punto a tener en cuenta, es que la segmentación perfecta de una imagen; es decir, cada pixel se asigna al segmento correcto, es un objetivo que generalmente no se puede lograr. Esto es debido a la forma en que una imagen digital se adquiere, ya que un pixel puede estar sobre el límite "real" entre objetos, así que pertenece parcialmente a dos objetos.

Segmentación basada en umbrales

El umbral por definición es probablemente la técnica más utilizada para segmentar una imagen. La operación de umbralización inicialmente fue una operación de reasignación de valor de gris g definida por la expresión (3).

$$g(v) = \begin{cases} 0 & if \quad v < U \\ 1 & if \quad v \geq U \end{cases} \tag{3}$$

Donde v representa un valor gris y U es el valor del umbral. La umbralización transforma una imagen de valores grises a una imagen binaria. Así la imagen se divide en dos segmentos, identificados por los valores de pixel 0 y 1, respectivamente. Por ejemplo, si se tiene una imagen que contiene objetos brillantes sobre un fondo oscuro, la umbralización se puede usar para segmentar los objetos, como se observa en la Figura 2.7.

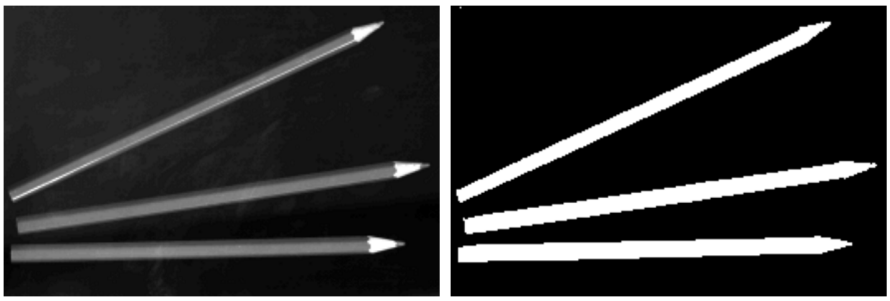


Figura 2.7 Ejemplo de segmentación por umbralización

Es importante tener en cuenta que el valor del umbral se suele calcular para cada imagen basándose en el histograma de cada una de éstas. Pero como los videos de la base de datos que se van a utilizar fueron grabados bajo condiciones controladas de iluminación y ubicación de los objetos de interés, los umbrales de segmentación fueron previamente definidos por los mismos autores de la base de datos, ya que se conoce previamente la ubicación de la superficie de interés por medio de la dimensión de profundidad de los videos RGB-D, permitiendo eliminar el fondo para facilitar el reconocimiento a realizar. Posteriormente, también es posible eliminar otros elementos de la imagen por segmentación de color ya que siempre se utilizó el escenario bajo las mismas condiciones de iluminación.

2.4.2 Flujo óptico

El término flujo óptico está relacionado con el patrón de movimiento aparente en una escena visual, causada por el movimiento relativo entre un observador y una escena. En este punto es importante resaltar que este concepto fue introducido por el psicólogo James J. Gibson en la década de 1940, resaltando la importancia del flujo óptico para la percepción de los ofrecimientos dados como la capacidad de percibir las posibilidades de acción dentro del entorno.

En el campo de la visión artificial, el término flujo óptico se ha utilizado para relacionar técnicas del procesamiento de imágenes como detección de movimiento o control de la navegación (Royden & Moore, 2012). Éste permite consignar en una sola imagen la información temporal de una secuencia, ya que permite la correspondencia de la información en dos imágenes consecutivas, realizándose normalmente de 2 formas:

- Disperso: cuando se utilizan características extraídas de las imágenes.
- Denso: cuando se tienden a utilizar todos los pixeles de la imagen.

El flujo óptico intenta calcular el movimiento entre dos fotogramas que se toman en los momentos t y $t + \Delta t$. Para el caso de una imagen un pixel en la ubicación (x, y, t) con intensidad $I(x, y, t)$ se habrá movido $\Delta x, \Delta y$ y Δt entre los dos marcos de imagen y adicionalmente se puede aplicar una restricción de constancia de brillo donde $I(x, y, t) = I(x + \Delta x,\ y + \Delta y,\ t + \Delta t)$.

Suponiendo un valor pequeño, del desplazamiento de la ubicación de un punto de un fotograma con respecto a la ubicación del mismo punto en el fotograma siguiente, se genera una restricción de la imagen en I(x, y, t), con la serie de Taylor puede desarrollarse para obtener la expresión (4).

$$I(x + \Delta x,\ y + \Delta y,\ t + \Delta t) = I(x, y, t) + \frac{\partial I}{\partial x}\Delta x + \frac{\partial I}{\partial y}\Delta y + \frac{\partial I}{\partial t}\Delta t + \cdots \quad (4)$$

De estas ecuaciones se deduce la expresión (5).

$$\frac{\partial I}{\partial x}\Delta x + \frac{\partial I}{\partial y}\Delta y + \frac{\partial I}{\partial t}\Delta t = 0 \quad (5)$$

Y dividiendo (4) por Δt cómo se ve en la expresión (6).

$$\frac{\partial I}{\partial x}\frac{\Delta x}{\Delta t} + \frac{\partial I}{\partial y}\frac{\Delta y}{\Delta t} + \frac{\partial I}{\partial t}\frac{\Delta t}{\Delta t} = 0 \quad (6)$$

Se obtiene la expresión (7).

$$\frac{\partial I}{\partial x}Vx + \frac{\partial I}{\partial y}Vy + \frac{\partial I}{\partial t} = 0 \quad (7)$$

Donde Vx, Vy son las componentes X y Y de la velocidad o flujo óptico de I(x,y,t) y $\frac{\partial I}{\partial x}, \frac{\partial I}{\partial y}, \frac{\partial I}{\partial t}$ son las derivadas de la imagen en (x,y,t) en las direcciones correspondientes. Ix, Iy y It se pueden escribir para las derivadas a continuación.

$$Ix\,Vx + Iy\,Vy = -It \qquad \text{o} \qquad \nabla I^T \cdot \underset{V}{\rightarrow} = -I\,t \tag{8}$$

La expresión (8) muestra una ecuación con dos incógnitas y no se puede resolver como tal. Esto se conoce como el problema de apertura de los algoritmos de flujo óptico. Para encontrar el flujo óptico, se necesita otro conjunto de ecuaciones, dado por alguna restricción adicional. Todos los métodos de flujo óptico introducen condiciones adicionales para estimar el flujo real. Dentro de los diferentes métodos de estimación del flujo óptico basados en derivadas parciales de la señal de imagen y / o el campo de flujo buscado y derivadas parciales de orden superior, tales como:

- Método de Lucas-Kanade: con respecto a los parches de imágenes y un modelo afín para el campo de flujo (Zhang & Chanson, 2018).
- Método de Horn-Schunck: optimización funcional basado en los residuos de la restricción de constancia de brillo, y un término de regularización particular que expresa la suavidad esperada del campo de flujo (Zhang & Chanson, 2018).
- Método de Buxton-Buxton: basado en un modelo del movimiento de los bordes en secuencias de imágenes (Humphreys & Bruce, 1989).
- Método Black-Jepson: flujo óptico grueso a través de la correlación (Beauchemin & Barron, 1995).
- Métodos generales variacionales: una gama de modificaciones o ampliaciones de Horn-Schunck, utilizando otros términos de datos y otros términos de suavidad.

En esta tesis se emplea la función `estimateFlow` de Matlab para calcular el flujo óptico entre cada par de imágenes consecutivas como se observa en la Figura 2.8. Adicionalmente se utiliza el método de Horn-Schunck que introduce una restricción global de suavidad para resolver el problema de apertura. Las ventajas del algoritmo Horn-Schunck incluyen que produce una alta densidad de vectores de flujo, es decir, la información de flujo que falta en las partes internas de los objetos homogéneos se rellena desde los límites de dicho objeto que se ha desplazado. Como punto negativo se tiene que es más sensible al ruido que los métodos locales.

Figura 2.8 Ejemplo de flujo óptico, fotograma en un tiempo inicial, fotograma un tiempo después, campo vectorial del desplazamiento de cada pixel de la imagen.

2.5 Resumen

Para esta sección de marco teórico se describieron los conceptos más importantes para el desarrollo del proyecto, empezando por los ofrecimientos de los objetos, donde además de presentar la idea tras este concepto se presentaron los métodos de aprendizaje por experimentación y por demostración que se habían presentado en investigaciones previas. También se presentó el concepto de Red Bayesiana (BN), donde se habló de representación e inferencia de este y el concepto de Red Neuronal Convolucional (CNN), donde se habló de las arquitecturas y diferentes tipos de capas de las que se compone una red. Por último, se habla del procesamiento de imágenes y videos, presentando conceptos específicos como segmentación de imágenes, Flujo óptico.

En el siguiente capítulo se tiene el trabajo relacionado donde se presentan diversas investigaciones de diferentes grupos que tienen relación con este proyecto. Básicamente se muestra la evolución del concepto de los ofrecimientos de los objetos desde su concepción en el campo de la psicología perceptual, pasando por diferentes modelos y ayudando en la elaboración de diferentes tareas, hasta llegar a ser un punto de interés en la selección de la arquitectura de CNNs enfocadas al reconocimiento de objetos y acciones.

Capítulo 3

Trabajo relacionado

En este capítulo se presentan los trabajos de investigación más relacionados con el desarrollo del proyecto. Partiendo de la introducción del concepto de los ofrecimientos en 1979 por el psicólogo perceptual J.J Gibson. Después se pasa a revisar cómo se han modelado y usado los ofrecimientos en el campo de la robótica y las ciencias computacionales, donde las relaciones entre objetos, acciones y efectos se han establecido mediante el uso de redes bayesianas que permiten llevar a cabo tareas de inferencia para imitar o predecir comportamientos de seres humanos. Llegando finalmente al uso de los ofrecimientos en la arquitectura de CNNs, donde se plantea que las relaciones mencionadas anteriormente son útiles al aportar información adicional en tareas de reconocimiento tanto de acciones como de objetos.

3.1 Ofrecimientos

Los "ofrecimientos" de los objetos fueron presentados como "Posibles Acciones" por el psicólogo perceptual en su libro seminal (Gibson, 1978) más específicamente en el capítulo 8 donde presenta *"The Theory of Affordances"*. Se plantea que la percepción de los objetos no solamente está definida por sus características visuales, sino que adicionalmente, éstas nos dan indicios de su funcionalidad, es decir de las acciones que se pueden llevar a cabo con un objeto, complementando así la percepción del mismo. Posteriormente este concepto fue introducido en el campo de la Interacción Humano-Computadora (HCI, *Human-Computer Interactions*) por (Norman, 1988), donde se presentaban los ofrecimientos como un aspecto relacionado con el diseño del objeto, que sugiere la función o uso del mismo.

3.1.1 Modelado y Uso directo de los ofrecimientos

Esta subsección se dividirá en 4 partes, presentando así las investigaciones desarrolladas por diferentes equipos de trabajo, que han modelado y realizado diversas tareas, relacionadas con los ofrecimientos. Partiendo del "Instituto de Sistemas y Robótica" en Lisboa donde se trabajó este tema en 2007 y 2008, siguiendo con el "Departamento de Ciencias de la Computación" en Leuven entre los años 2012 y 2014, pasando al también "Departamento de Ciencias de la Computación" pero en Cornell entre 2012-2014, hasta llegar a "Escuela de Computación Interactiva" en Georgia donde se trabajó esta temática entre 2014 y 2017.

Instituto de Sistemas y Robótica, Lisboa (2007-2008)

Para empezar, se presenta el equipo del instituto superior técnico de Lisboa, Portugal, liderado por Luis Montesano y Manuel Lopes, quienes en (Montesano, et al., 2007) fueron los primeros en utilizar los ofrecimientos de los objetos en el campo de las ciencias computacionales, representando el comportamiento de los objetos en términos de las habilidades motoras y de percepción con que cuenta un robot. Después, en (Montesano, et al., 2007), se combinó la coordinación percepto-motriz con el aprendizaje de las relaciones estadísticas entre acciones y propiedades del objeto, para realizar simples juegos de imitación que proporcionan interpretación y planificación de tareas. También se estableció en (Lopes, et al., 2007) una estructura, donde, a partir de la interacción con el mundo se aprenden los ofrecimientos, luego se observa una demostración de la cual se extrae la transición en el mundo (dependiente de la tarea), para así interpretar la demostración y decidir cómo se puede imitar.

Ese trabajo de varios años queda consolidado en (Montesano, et al., 2008) donde se tiene un robot dotado con un conjunto de habilidades tanto de percepción como de movimiento. Partiendo de la idea de que el robot era capaz de detectar objetos cercanos y medir características básicas como su posición, color, forma o tamaño y además podía interactuar con los objetos, de una manera simple a través de un repertorio de acciones predefinidas. De este modo los ofrecimientos surgieron cuando el robot aplicó sus acciones $A = \{a_1, a_2 \dots, a_n\}$ a diferentes objetos $O = \{o_1, o_2 \dots, o_n\}$ circundantes y observó los efectos $E = \{e_1, e_2 \dots, e_n\}$ resultantes. Para el modelado de los ofrecimientos, el objetivo fue aprender las relaciones estadísticas entre las acciones y las propiedades de los objetos. Se propuso un modelo general basado en una red bayesiana como se observa en la Figura 3.1 a) donde se vinculan las acciones

en la parte superior derecha de la imagen, con las características de los objetos a la izquierda y las características de los efectos en la parte inferior. Posteriormente, la estructura de la red se aprendió de la experimentación, donde, el robot fue presentado con un objeto y éste seleccionó al azar una acción a realizar. El aprendizaje de la estructura del grafo se logró estimando la distribución en las posibles estructuras de red dados los datos, usando *(MCMC, Markov Chain Monte Carlo)* de (Heckerman, et al., 1995), se obtiene la red de la Figura 3.1 b).

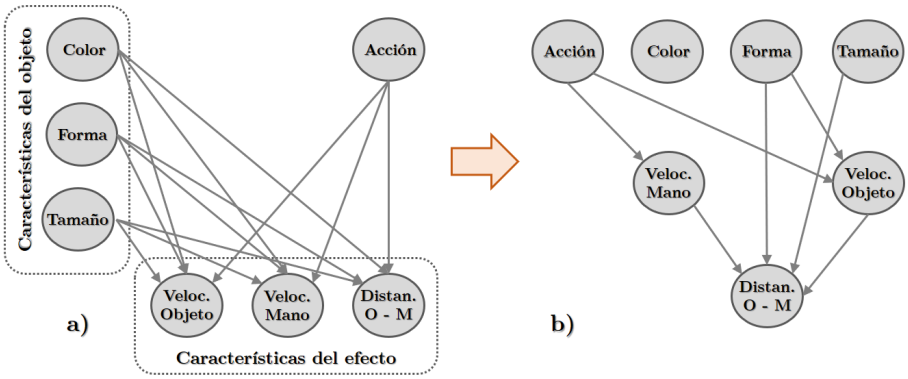


Figura 3.1 Estructura de red bayesiana presentada por Luis Montesano. a) Estructura inicial. b) Estructura aprendida de la red

Con la red ya aprendida se pasa a la tarea de imitación, donde primero el robot observa a un humano actuando sobre un objeto como se observa en la Figura 3.2.a) y debe percibir la acción presentada. Ahora puede copiar el comportamiento o tratar de inferir las partes importantes de la tarea para interpretar las acciones realizadas por otros en términos de las propias acciones del agente; es decir, deben hacer coincidir los efectos. Esto está directamente relacionado con la métrica utilizada ya que el robot puede elegir una acción diferente (al compararse con la realizada por el maestro) siempre que su experiencia le indique que se puede alcanzar el efecto deseado.

La métrica de imitación se puede definir de varias formas para lograr diferentes comportamientos en un juego de imitación. Por un lado, para imitación sólo del efecto, el objetivo de este comportamiento es lograr el mismo efecto que el observado, actuando sobre el objeto que más fácilmente produzca el mismo efecto, la recompensa no depende del objeto como se observa en la Figura 3.2.c) donde el robot decide manipular la esfera amarilla pequeña (porque es más fácil de agarrar), aunque la

demostración se realizó con el cubo azul. Por otro lado, para la imitación del efecto y el objeto, la métrica agrega información sobre las características del objeto en la función de costo, esto permite favorecer aquellos objetos similares a los manipulados por el maestro, Figura 3.2.d) donde se toma el cubo azul, al igual que en la demostración. Por último, la planeación se toma como un problema de decisión bayesiano de un paso P(A|E,O).

Figura 3.2 Aprendizaje de tarea de imitación basada en los ofrecimientos (Montesano, et al., 2008). a) Demostración de la tarea a realizar. b) El robot se enfrenta a la escena. c) Imitación basada en repetir solo la acción. d) Imitación basada en repetir la acción con el mismo objeto.

Este trabajo es de gran importancia para nuestra tesis, ya que ellos fueron los primeros en utilizar el concepto de los ofrecimientos de los objetos en robótica, además modelaron dichos ofrecimientos con una BN al igual que lo planteado en nuestra metodología. Pero a diferencia de nuestra tesis, Montesano y su equipo sólo contaban con 3 tipos de acciones, definidas por una sola variable y 2 tipos de objetos en tamaños diferentes definidos a partir de las variables color, forma y tamaño. Mientras nosotros contamos con 14 tipos de objetos, definidos por una sola variable discreta llamada (objeto) y 13 tipos de acciones, también definidas por una sola variable discreta llamada (acción). Adicionalmente, nosotros realizamos tareas de inferencia que tienen dos enfoques distintos; por un lado, tenemos el mejoramiento de los reconocimientos de acciones objetos y efectos, cuando se cuenta con información de todas las variables; por el otro lado, podemos estimas una variable en función a las otras dos. Mientras que el trabajo de Montesano se limitó a realizar tareas de imitación.

Departamento de Ciencias Computacionales, Leuven(2012 - 2014)
Posteriormente, otro equipo encabezado por Bogdan Moldovan y Luc De Raedt, en la Katholieke Universiteit Leuven, Belgium, presentó en (Moldovan, et al., 2011) una extensión del modelo, utilizando aprendizaje relacional estadístico (SRL), combinando un conjunto de hechos probabilísticos con reglas lógicas en dominios robóticos (lenguajes de programación probabilísticos).

Este nuevo enfoque se utilizó en (Moldovan, et al., 2012) para modelar los ofrecimientos, aprendiendo de la interacción robótica en escenarios con múltiples objetos que interactúan entre ellos. Esto aumenta las capacidades cognitivas básicas, lo cual permitió pasar de interacciones de dos objetos a una cantidad variable de objetos, adicionando así conceptos de alto nivel. De forma similar a (Montesano, et al., 2008) se partió de unas habilidades básicas del robot para construir, en primer lugar, habilidades motoras que permitieron realizar las acciones y en segundo lugar, habilidades perceptivas para medir; por un lado, las características como la segmentación del color y localización 3D de los objetos; por otro lado, los efectos producidos, medidos como las diferencias en los atributos del objeto antes y después de que se realiza la acción.

En una primera fase, se aprendió la estructura de la BN, utilizando el algoritmo K2 y en la segunda fase, se aprendieron los parámetros. Después, se pasó a la ejecución de tareas donde el robot observa la escena como en la Figura 3.3.a) y conoce el efecto que desea generar (poner el objeto morado en la región indicada como se aprecia en la Figura 3.3.b). En este punto, el robot simulado intenta inferir la acción a realizar vinculando los modelos relacionales para la extensión de la tarea de un solo objeto a configuraciones más generales de manipulación que pueden involucrar múltiples objetos, usando un modelo relacional probabilístico, donde las relaciones incluían la distancia relativa entre los objetos, su ángulo de orientación y contacto. Adicionalmente, ProbLog se incluyó como un lenguaje de programación probabilístico (PPL), éste específicamente diseñado para describir y razonar con modelos relacionales probabilísticos. Finalmente, se demostró que el modelo SRL obtenido de las interacciones con dos objetos se pudo usar como ajuste general para escena con más de dos objetos, lo que indicó que las interacciones entre tres o más objetos no necesitan ser explícitamente consideradas, para el tipo de tareas en ese trabajo.

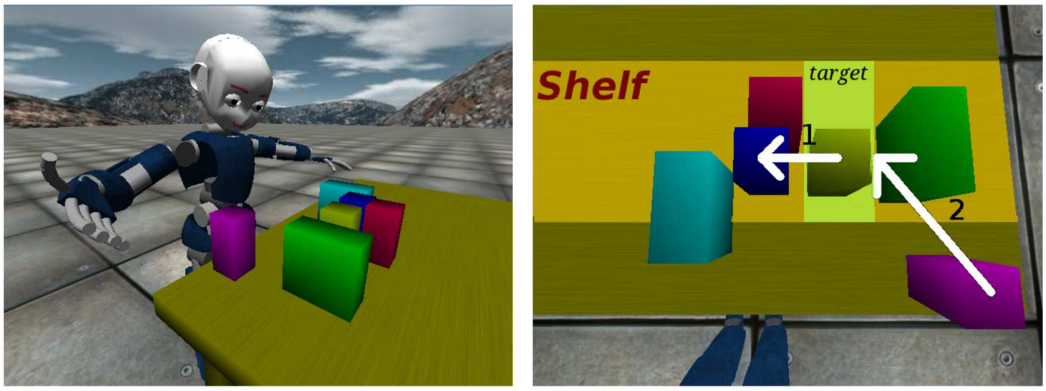


Figura 3.3 a) Escenario de estantería, b) secuencia de acciones para colocar un objeto adicional (Moldovan, et al., 2012)

Posteriormente, este mismo equipo presentó tres extensiones de los ofrecimientos relacionales. Primero en (Moldovan, et al., 2013) se realizó una adaptación del modelo probabilístico relacional para desempeñar tareas secuenciales de manipulación robótica. Esto se desarrolló en dos partes. Por un lado, se emplearon modelos de accesibilidad secuencial para reconocer las acciones individuales que componen una demostración de alto nivel. Por el otro lado, se utilizaron los modelos de conceptos para planificar un curso de acción adecuado para replicar los efectos observados en la demostración, para esto se adoptó el enfoque de decisión relacional de los procesos de Markov.

Segundo, se utilizaron los ofrecimientos relacionales como ayuda en la búsqueda de objetos ocultos en (Moldovan & Raedt, 2014-a). Su enfoque se basó en la idea de mejorar el rendimiento de la búsqueda, ya que se puede establecer cuál de los múltiples objetos permite una acción dada, en función a sus distribuciones de probabilidad dependiente de su forma y tamaño.

Tercero, en (Moldovan & Raedt, 2014-b) se adaptaron los ofrecimientos relacionales para usarlos en manipulaciones de robots de dos brazos. Para empezar, en este caso la BN presenta una distribución que mezcla variables discretas con variables continuas, que se aprende inicialmente en una etapa de balbuceo comportamental y luego con el uso del aprendizaje relacional estadístico. Después se construyó un modelo simétrico para el brazo adicional, así, en la manipulación con dos brazos las acciones se pueden modelar en un esquema donde los brazos pueden actuar de forma secuencial o simultánea.

El enfoque presentado por este grupo es relevante ya que principalmente permite la interacción entre varios objetos al combinar los hechos probabilísticos con las reglas lógicas. Pero a diferencia de nosotros los objetos con los que trabajaron Moldovan y su grupo son básicamente bloques por lo tanto las acciones se limitan a agarrar y soltar en un punto determinado, desaprovechando así las amplias capacidades de los ofrecimientos. De este modo se pudo apreciar que la tarea realizada por ellos es básicamente una planeación relacional de agarres de los bloques. Mientras nosotros realizamos tareas de mejoramiento del reconocimiento o inferencia aplicables en planeación de acciones, selección de objetos o predicción de efectos. Esta variedad de tareas está relacionada; por un lado, con el modelo utilizados para representar los ofrecimientos que es una BN; por el otro lado está relacionado con la variedad de objetos y acciones posibles dentro de las que encontramos esponjas que se pueden exprimir o brochas con las que se puede pintar entre otras más con las cuales sí se cuenta en la base de datos utilizada en nuestro proyecto de tesis.

Departamento de Ciencias Computacionales, Cornell (2012-2014)

Este equipo de Cornell University de New York, representado por Hema Swetha Koppula y Ashutosh Saxena Ithaca, consideró el problema de etiquetar conjuntamente las posibilidades de objetos y las actividades humanas de videos RGBD. En (Koppula, et al., 2012) y (Koppula, et al., 2013), se presentó el problema como un campo aleatorio de Markov, para el cual los nodos representan objetos y subactividades, y los bordes representan las relaciones entre los ofrecimientos de objetos, sus relaciones con las subactividades y su evolución a través del tiempo. La idea clave del trabajo fue notar que, en la detección de actividades, a veces es más informativo saber cómo se usa un objeto (ofrecimientos asociados) en lugar de saber qué objeto es (categoría del objeto).

Después, este mismo equipo aplicó su enfoque en la comprensión funcional de un entorno, en términos de las actividades y los objetos (Koppula & Saxena, 2014), donde la descripción de los objetos que admiten diversas acciones representan la interacción de un sujeto con el medio. Tal entendimiento es útil para muchas aplicaciones, tales como la detección de actividad y robótica asistida, como en (Koppula, 2015), donde se presenta la anticipación de actividades que una persona hará a continuación, un aspecto importante de la percepción humana. Esto puede ser utilizado por un robot de asistencia para planificar con anticipación las respuestas en los entornos humanos. En el trabajo citado, se presentó un enfoque constructivo

para generar varias actividades futuras posibles, razonando sobre las relaciones espacio-temporales con ayuda de las posibilidades de los objetos. Se representó cada posible futuro usando un campo aleatorio condicional temporal anticipatorio (ATCRF), donde se probaron los nodos y los bordes correspondientes a las trayectorias futuras de los objetos y poses humanas desde un modelo generativo. También en (Koppula & Saxena, 2013) se observó una escena que contiene un ser humano y objetos para un tiempo t en el pasado y se anticiparon las posibilidades futuras. El objetivo fue calcular una distribución sobre los posibles estados futuro de subactividad, poses humanas y ubicaciones de objetos, como se observa en Figura 3.4 donde a la derecha se muestran en negro las posibles trayectorias de las acciones futuras a realizar con el objeto de interés.

Figura 3.4 Distribución de futuras posibles acciones (Koppula & Saxena, 2013), a) escena donde una persona va a interactuar con una taza, b) posibles trayectorias de interacción en el futuro.

El enfoque presentado en estas investigaciones tiene como aporte principal a nuestro proyecto el aprendizaje de los ofrecimientos a partir de demostraciones, ya que en el trabajo desarrollado por Koppula y Saxena no se cuenta con un robot que ejecute las acciones, sino que se tiene un agente el cual aprende los ofrecimientos a partir de la percepción de las interacciones entre un llamado “maestro” (quien realiza la acción, siendo independiente de quien lo observa) y los objetos presentes en el entorno. La mayor diferencia con nuestro proyecto radica en que ellos se enfocaron en la tarea de predecir las acciones que se van a realizar. Mientras nosotros realizamos diferentes grupos de tareas como mejora del reconocimiento e inferencia las cuales serán útiles en tareas relacionadas con la manipulación robótica.

Escuela de Computación Interactiva, Georgia (2014-2017)

Más recientemente las investigadoras Vivian Chu y Andrea L. Thomaz, del Georgia Institute of Technology de Atlanta, desarrollaron algunos trabajos relacionados con los ofrecimientos. Inicialmente se interesaron en comprender la función que desempeña el tacto en los ofrecimientos (Chu & Thomaz, 2014), donde se basaron en que los humanos manipulan y aprenden sobre los objetos no sólo usando visión, sino también una entrada sensorial física del tacto. En la investigación citada se caracterizó fuerzas y torque en el sensor (FTS) montado en un robot mientras realiza cinco tareas individuales, enfocados en analizar interacciones exitosas y fallidas.

Después se enfocaron en el aprendizaje de los ofrecimientos táctiles, donde el objetivo era que el robot percibiera la fuerza y el torque al realizar una tarea, esto basado en demostración y exploración guiada por humanos (Chu & Thomaz, 2016) (Chu, et al., 2016). Aquí una demostración consiste en que una persona mueve los actuadores del robot a ciertas posiciones deseadas y se espera que el agente tome una lectura de sus sensores, como se observa en la Figura 3.5, donde se aprecia una persona llevando al robot por las diferentes etapas de la acción y así, se pueden aprender rápidamente varias acciones primitivas.

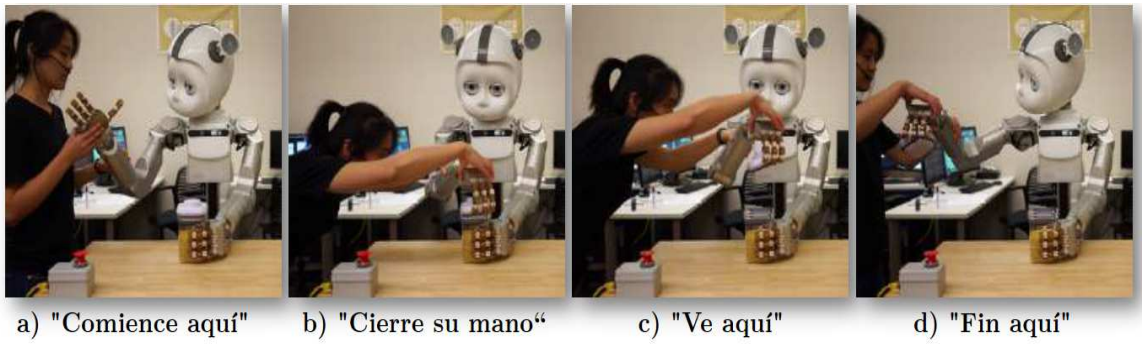


Figura 3.5 Ejemplo de aprendizaje de ofrecimientos táctiles por demostración (Chu, et al., 2016)

La exploración guiada por humanos, consiste en que el robot intenta repetir la acción demostrada varias veces como se observa en la Figura 3.6, pero adicionalmente el humano mueve el objeto un poco en cada intento, para perturbar ligeramente la acción y que el aprendizaje sea más robusto. Este tipo de enseñanza es conocida como “andamios ambientales” y proporciona un aprendizaje de alta calidad, ya que se reduce en enorme medida la exploración de un espacio de búsqueda muy grande.

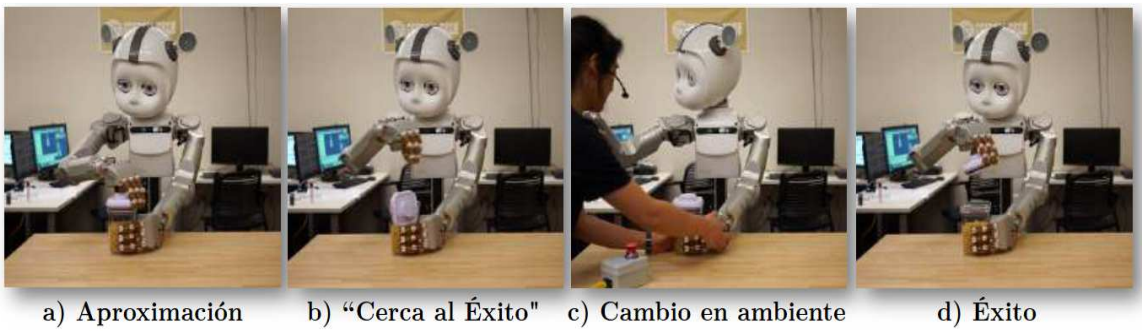


Figura 3.6 Ejemplo de aprendizaje de ofrecimientos táctiles por exploración guiada (Chu, et al., 2016)

Por último, las investigadores enfocaron sus esfuerzos en establecer la diferencia entre el aprendizaje por autoexploración y cuando se tiene exploración guiada por un ser humano, (Chu, et al., 2016) y (Chu & Thomas, 2017). La importancia del enfoque presentado radica en que la autoexploración puede llegar a ser un proceso muy lento ya que se debe hacer un barrido sobre todo el espacio de muestreo; mientras que en la exploración guiada, los maestros humanos se centran naturalmente en mostrar los aspectos más destacados de los objetos al proporcionar demostraciones. Por lo tanto, el aprendizaje por exploración guiada es más rápido que la autoexploración y además está enfocado en lo que realmente es importante.

Al comparar el enfoque de estos trabajos con nuestro proyecto de tesis, se presentan algunas diferencias bien marcadas. Para empezar Chu y Thomaz utilizan los sensores táctiles como una de las variables a percibir y también se enfocan en establecer un modelo de enseñanza al robot para que este aprenda a ejecutar las diferentes tareas. Mientras que en nuestro desarrollo no contamos con un robot, por lo tanto, nos enfocamos en el modelo que relaciona acciones, objetos y efectos, realizando tareas de percepción relacionada con el reconocimiento de las variables. Pero se debe tener en cuenta que más adelante planeamos utilizar los modelos aprendidos en tareas de ejecución de interacciones con objetos, para las cuales utilizaremos la capacidad de inferencia de nuestro modelo, en tareas como la planeación de acciones, selección de objetos y predicción de los efectos.

3.2 Reconocimiento de objetos y acciones

Los reconocimientos tanto de objetos como de acciones son tareas de alto interés en las ciencias computacionales, gracias a sus diversas aplicaciones, por este motivo, se han diseñado diferentes metodologías para abordar estos problemas. En los últimos años las capacidades de reconocimiento han mejorado notoriamente con el uso de las CNNs, alcanzando así, niveles de acierto comparables con las habilidades humanas. Dentro de las diversas arquitecturas presentes en la literatura, nuestro interés se enfoca en las que consideramos tienen una relación ya sea directa o indirecta con los ofrecimientos de los objetos, que es un concepto fundamental en el desarrollo de este proyecto.

3.2.1 Uso directo de los ofrecimientos en el reconocimiento de objetos

La investigación desarrollada en (Thermos, et al., 2017) es muy importante para nosotros, ya que está directamente relacionada con nuestro proyecto de tesis, por varias razones. Para empezar, Thermos y su equipo son los autores de la base de datos CERTH-SORD3D, la cual utilizaremos para nuestro desarrollo. Además, presentan varias arquitecturas de CNNs compuestas básicamente de dos flujos, combinando la percepción de los objetos con la percepción de los ofrecimientos, algo similar a nuestra metodología. Por último, el texto citado se hace importante ya que compararemos nuestros resultados en la sección 5.3 con los obtenidos en la investigación de Thermos, teniendo en cuenta que en ese trabajo sólo realizan la tarea de reconocimiento de objetos.

Para entender la metodología planteada, se debe empezar por conocer lo que Thermos llama modelo de flujo único, que consiste en una CNN con una arquitectura basada en VGG-16 como la que se presenta en la Figura 3.7 la cual tiene 13 capas convolucionales (CONV) que se dividen en 5 bloques separados por capas de *max-pooling* que están seguidas de 3 capa completamente conectadas (*FC, Fully Connected*) y Después de cada capa CONV o FC, se tiene una unidad lineal rectificada *(RL, Rectified Linear Unit).*

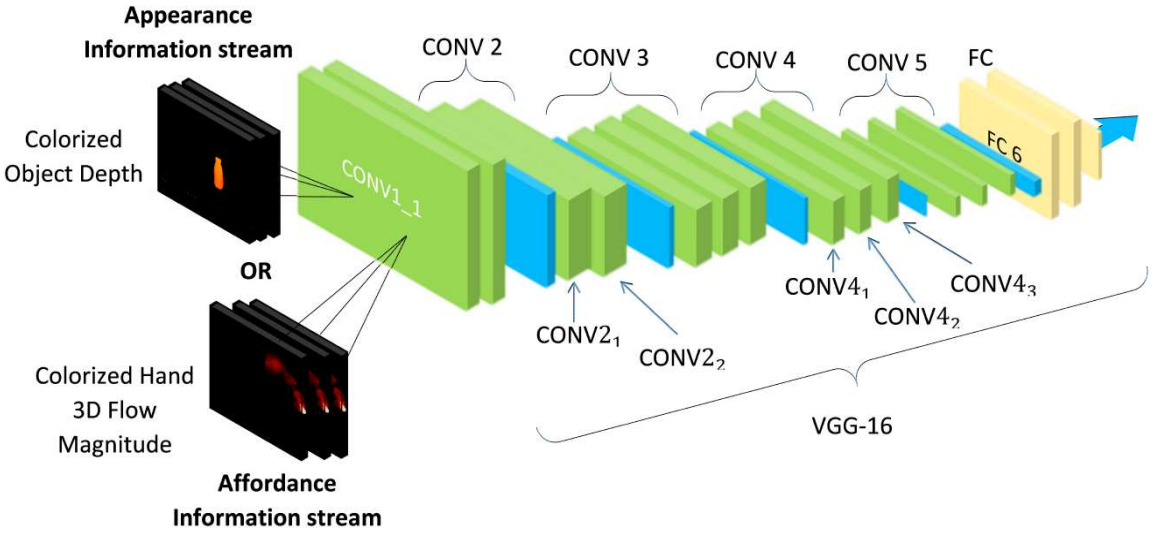


Figura 3.7 Modelo de flujo único con arquitectura VGG-16, con 13 capas convolucionales(verdes), 5 capas max-pooling(azules) y 3 capas completamente conectasa(amarillas) (Thermos, et al., 2017).

En la etapa inicial se calcula el porcentaje de acierto para dos redes de flujo único. La primera se alimentó con una imagen segmentada del objeto, para obtener el reconocimiento del objeto. La segunda se alimentó con una imagen del flujo óptico, para obtener lo que ellos llamaron el reconocimiento del ofrecimiento, que para nuestro trabajo estas serán las acciones. Los resultados obtenidos se presentan en la Tabla 3.1

Tabla 3.1 Resultados del reconocimiento de flujo único.

Método	**Tarea**	**(%)Acierto**
Apariencia CNN	Reconocimiento de objeto	85.12
Affordable CNN	Reconocimiento del ofrecimiento	81.92

Después se realizaron pruebas con varias CNNs que tomaban como entrada 20 fotogramas uniforme mente espaciados y luego combinaban la información de los objetos con la de los ofrecimientos en arquitecturas de dos flujos denominadas (*GST, Generalized Spatio-Temporal*), las cuales se presentan en la Figura 3.8, adicionalmente estas arquitecturas contaban con (*LSTM, Long-Short Term Memory*), las cuales de presentaban antes o después de la capa de fusión.

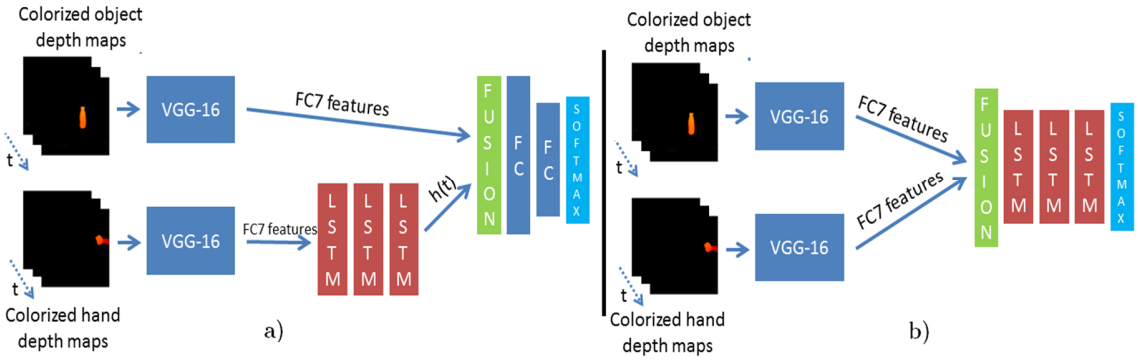


Figura 3.8 Arquitecturas de dos flujos GST, a) arquitectura con LSTM sólo para el flujo del ofrecimiento antes de la fusión, b) arquitectura con LSTM después de la capa de fusión

Con la idea mencionada anteriormente plantearon un conjunto de arquitecturas de red con diferentes tipos de fusión como la sincrónica tardía *(LS, Late Synchronous)*, la asíncrona tardía (LA, Late Asynchronous), la de nivel único lento (SSL, Slow Single Level) y la de nivel múltiple lento (SMLg, Slow Multi Levelg). Teniendo en cuenta, que la evaluación se realizó en dos escenarios; por un lado, cuando para la decisión final de clasificación de objetos se consideró la predicción de solo el último fotograma (último); por el otro lado, cuando se promediaron las predicciones de todos los fotogramas (todos). Los resultados obtenidos se presentan en la Tabla 3.2, donde se pueden apreciar diferentes mejoras en el porcentaje de acierto y la mayor mejora reportada es de 1.38%.

Tabla 3.2 Resultados de mejora del porcentaje de acierto con la arquitectura GST

Arquitectura basada en GST	(%) Acierto
GST *LS* (último)	86.28
GST *LS* (todos) [1 *CONV*, 2 *FC*]	**86.50**
GST *LA* (todos, T = 2)	86.42
GST *LA* (todos, T = 4) [1 *CONV*, 2 *FC*]	86.17
GST *LA* (todos, T = 6) [1 *CONV*, 2 *FC*]	85.28
GST *SSL* (todos) [1 *CONV*, 2 *FC*]	79.65

Después se realizaron otras pruebas, nuevamente combinando la información de los objetos con la de los ofrecimientos, pero esta vez en arquitecturas que se alimentaban con solo una imagen para cada uno de los dos flujos denominada coincidencia de plantillas generalizadas *(GTM, Generalized Template-Matching)*, como las que se presentan en la Figura 3.9.

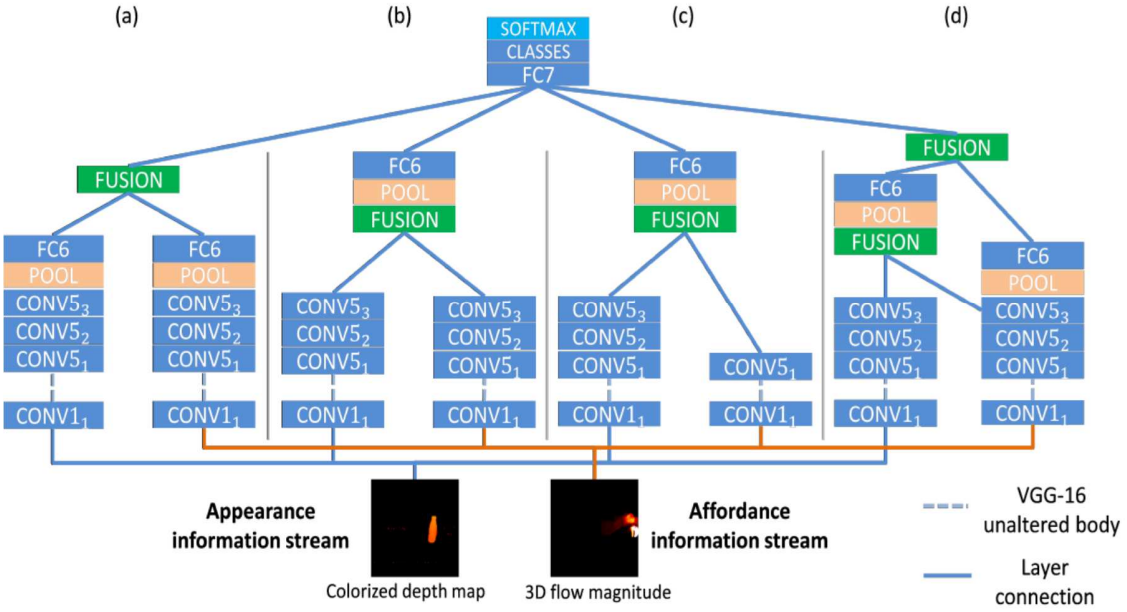


Figura 3.9 Arquitecturas de dos flujos GTM, a) GTM*LS*(FC6), b)GTM*LS*(RL53)[1CONV,1FC], c) GTM*LS*(RL53)[1CONV,1FC], d)GTM*SML*(RL5app3,RL5aff3,RL6)

Con la modificación mencionada se obtuvo un conjunto de resultados que se presentan en la Tabla 3.3 , donde se pueden apreciar algunas mejoras en el porcentaje de acierto de hasta 4.31%.

Tabla 3.3 Resultados de mejora del porcentaje de acierto con la arquitectura GTM

Arquitectura basada en GTM	(%) Acierto
GTM *LS* (FC6)	87.40
GTM *LS* (RL5 3) [1 CONV, 1 FC]	87.65
GTM *LS* (RL5 3) [1 CONV, 2 FC]	88.24
GTM *LS* (RL5 3) [2 CONV, 1 FC]	87.64
GTM *LS* (RL5 3) [2 CONV, 2 FC]	86.40
GTM *SSL* (RL3 *app3* , RL3 aff3)	78.74
GTM *SSL* (RL4 app3 , RL4 aff3)	87.20
GTM *SSL* (RL4 app3 , RL4 aff1)	85.82
GTM *LS* (RL5 3) [1 CONV, 1 FC]	88.13
GTM *SML* (RL5 app3 , RL5 aff1 , RL6)	88.23
GTM *SML* (RL5 app3 , RL5 aff3 , RL6)	89.43

Como se pudo apreciar, el enfoque presentado es parecido al nuestro, puesto que aquí también se busca enriquecer la información de entrada a la red para obtener una mejora en la capacidad de reconocimiento. Pero también se tienen diferencias, como que nosotros no nos vamos a limitar sólo al reconocimiento de los objetos, sino que también vamos a reconocer las acciones y los efectos. Otra diferencia importante radica en que nuestra metodología se basa en combinar CNNs con BN presenta flexibilidad ya que el modelo aprendido además de mejorar el reconocimiento puede ayudar en tareas de inferencia como selección de objetos, planeación de acciones y predicción de efectos.

3.2.2 Uso indirecto de los ofrecimientos en el reconocimiento de acciones

Para la tarea de reconocimiento de las acciones basado en CNNs, en la literatura se han investigado diversas arquitecturas, pero lo que llama nuestra atención en particular, son algunas arquitecturas en las cuales se combinan las características de los objetos con las características de las acciones, de modo similar a como lo hacen los ofrecimientos.

Un primer ejemplo de esto se puede ver en (Simonyan & Zisserman, 2014) donde se presenta una arquitectura de CNN de dos flujos (Espacial-Temporal) para el reconocimiento de acciones en videos. Estos dos flujos están estructurados como se observa en la Figura 3.10, donde: en el primer camino se tiene la información ***Espacial*** dada por un fotograma en el cual se pueden observar desde los objetos presentes en la escena hasta la pose del sujeto que realiza la acción y es procesado en una CNN de 7 capas. Por el otro camino se tiene la información ***Temporal*** dada por un apilamiento de los flujos ópticos del video en los cuales se consigna la información concerniente a los movimientos, dependientes de las acciones que se realizan, éste también procesado en una CNN de 7 capas. Por último, se combinan los puntajes de softmax por fusión tardía; esto significa que los caminos son independientes y solo comparten su información al final de la red. Un detalle importante a tener en cuenta es que esta red está diseñada estrictamente para reconocer acciones y no entrega ningún tipo de información final ni intermedia concerniente a los objetos, las poses o los efectos generados.

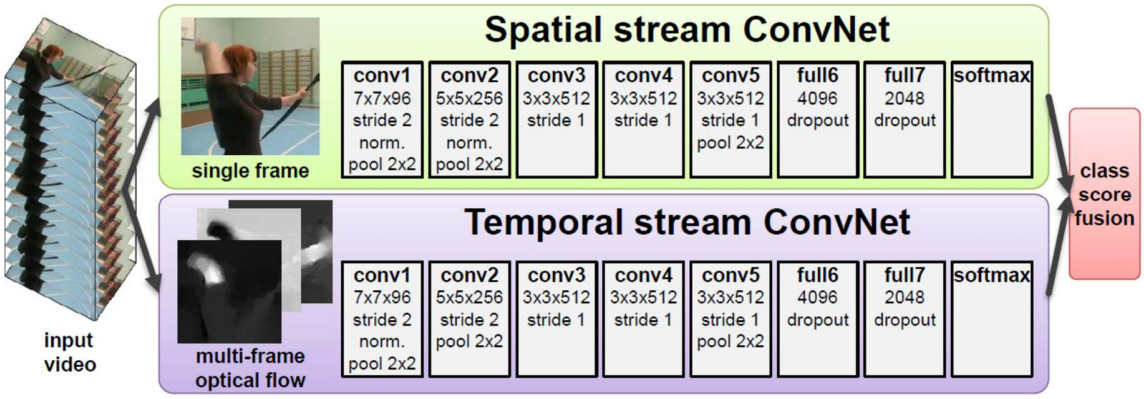


Figura 3.10 Arquitectura de dos flujos, *flujo espacial* en la parte superior y *flujo temporal* en la parte inferior (Simonyan & Zisserman, 2014).

Después de demostrar que la combinación de la información espacial con la temporal mejoraba la capacidad de reconocimiento, este mismo equipo trabajó en (Feichtenhofer, et al., 2016), donde el objetivo era encontrar el modo adecuado de combinar la información de los 2 caminos. Llegando a la conclusión de que es más útil fusionar las características espaciales y temporales en una capa de convolución sin pérdida de desempeño. También es mejor fusionar dichas redes en la última capa convolucional. Finalmente, la agrupación de características convolucionales abstractas sobre vecindarios espacio-temporales aumenta aún más el desempeño.

El trabajo citado, también está altamente relacionado con nuestra tesis ya que nuevamente el objetivo de la arquitectura planteada consiste en enriquecer la entrada de la red y de esta forma mejorar el reconocimiento, en este caso de las acciones. Pero nuevamente el enfoque presentado se limita a una única tarea ya que no se obtiene ningún tipo de información del reconocimiento de los objetos o los efectos y mucho menos se pueden realizar tareas de inferencia con la metodología propuesta basada en una CNN.

3.3 Resumen

En esta sección de trabajo relacionado se inició, con el trabajo de J.J. Gibson quien presentó la teoría de los ofrecimientos de los objetos. Después se presentaron diferentes modelos y usos directos de los ofrecimientos en el campo de las ciencias computacionales. Luego se resaltan 2 trabajos de gran influencia en nuestro proyecto en los cuales se utiliza el concepto de los ofrecimientos de los objetos, para determinar la arquitectura de CNNs utilizadas en tareas de reconocimiento de objetos y acciones

respectivamente. Nuestra metodología se destaca si tenemos en cuenta que los trabajos relacionados se enfocan en una sola tarea específica de reconocimiento ya sea de acciones, objetos o efectos. Por lo tanto, no se obtiene información de ninguna de las otras variables relacionadas en la interacción. Aquí se hace relevante nuestro proyecto, ya que al utilizar CNNs se acerca a los porcentajes de acierto del estado del arte en tareas de reconocimiento; y al combinar dichas CNNs con una BN, lo cual no se había hecho antes, la metodología desarrollada no se limita a tareas de reconocimiento ya que la BN también se puede utilizar en tareas de inferencia útiles en la manipulación robótica como: selección de objetos, planeación de acciones y predicción de efectos.

En el siguiente capítulo se presenta la metodología propuesta, especificando de forma más detallada cada uno de los componentes y procesos aquí realizados. Definiendo, por ejemplo: los umbrales de la segmentación realizada, las arquitecturas y parámetros de las CNNs y de la BN utilizada.

Capítulo 4

Método Propuesto

Para el desarrollo de este proyecto se ha planteado una metodología compuesta de 3 etapas como se observa en la Figura 4.1. En la primera etapa se tiene el preprocesamiento, donde a partir del video de una interacción entre una persona y un objeto se obtienen 3 imágenes, primero un fotograma segmentado el cual entrega la información relacionada al objeto y después dos flujos ópticos, uno que entrega información de la acción y el otro del efecto. Para la segunda etapa se tiene el reconocimiento inicial, dado como P(O'), P(A') y P(E'), los cuales se obtienen de tres CNNs alimentadas por las imágenes calculadas en la etapa anterior. Por último, en la tercera etapa se tiene una BN la cual fusiona la información de los 3 reconocimientos anteriores para obtener un "reconocimiento mejorado" P(O|O'A'E'), P(A|O'A'E') y P(E|O'A'E').

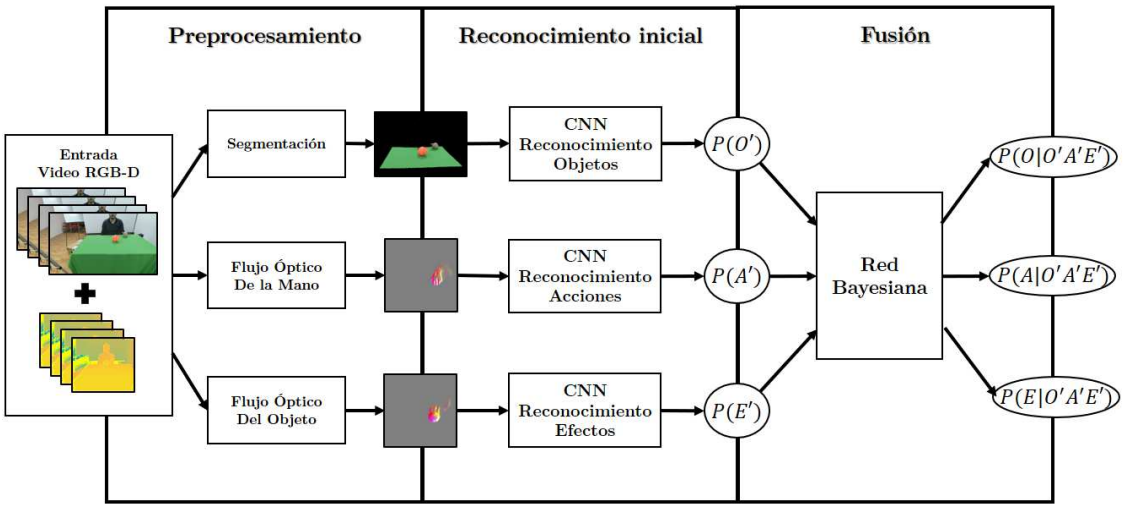


Figura 4.1 Diagrama de bloques de la metodología propuesta, compuesta de 3 procesos: preprocesamiento, reconocimiento inicial y fusión; para obtener estimaciones de objeto acción y efecto de los videos RGB-D de entrada.

4.1 Entrada

Como entrada al sistema se utilizan los videos de la base de datos CERTH-SOR3D presentados en (Thermos, et al., 2017), el uso de ésta se justifica basado en tres aspectos. Primero, el tiempo para desarrollar un proyecto a nivel de maestría es relativamente corto para elaborar una base de datos y esto no es una tarea sencilla. Segundo, esta base de datos está disponible para uso académico además de estar muy bien elaborada a comparación de otras que se encuentran en la literatura, ya que cuenta con una buena cantidad de sujetos, objetos y acciones. Tercero, es posible comparar la metodología propuesta con otras que se han presentado en la literatura, ya que esta base de datos está diseñada para ser un punto de referencia en el campo de los ofrecimientos de los objetos y ha sido utilizada en otras investigaciones.

4.1.1 Base de datos CERTH-SOR3D

El conjunto de datos elaborado por el *Information Technologies Institute* del *Centre for Research & Technology-Hellas* http://sor3d.vcl.iti.gr/. Está compuesto de un poco más de 20000 videos RGB-D (videos RGB 1920×1080 pixeles + secuencias de mapas de profundidad de 512×424 pixeles) de interacciones humano-objeto. El proceso de captura implicó tres sensores sincronizados de Microsoft Kinect v2, orientados como se observa en la Figura 4.2. Se debe aclarar que una interacción entre una persona y un objeto produce tres sujetos de la base de datos, uno para cada punto de vista. También se debe tener en cuenta que las capturas se realizan bajo condiciones ambientales controladas, todas las interacciones fueron realizadas en el mismo laboratorio siempre sobre la misma mesa con un mantel verde, para facilitar los posteriores procesos de segmentación.

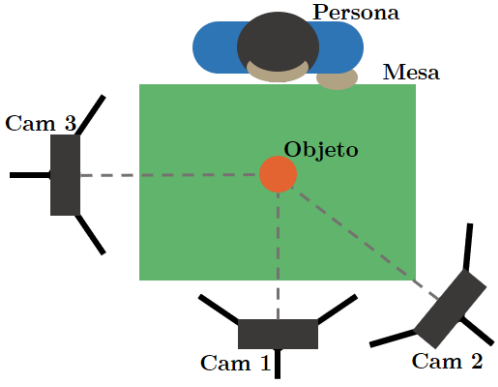


Figura 4.2 Esquema de captura, donde tres cámaras tipo *Kinect* captan la interacciones entre una persona y un objeto

La base de datos CERTH-SOR3D, originalmente está compuesta por 14 tipos de objetos y 13 posibles acciones que se presentan en dos listas que se incluyen a continuación.

Objetos

1. Pelota
2. Libro
3. Botella
4. Caja
5. Brocha
6. Lata
7. Pocillo
8. Martillo
9. Llave
10. Cuchillo
11. Bolígrafo
12. Jarra
13. Teléfono inteligente
14. Esponja

Acciones

1. Cortar
2. Agarrar
3. Martillar
4. Levantar
5. Abrir
6. Pintar
7. Verter
8. Empujar
9. Girar
10. Exprimir
11. Digitar
12. Desbloquear
13. Escribir

Los objetos y las acciones mencionadas se relacionan en sólo 54 interacciones, esto se debe a que algunos ofrecimientos son propios de un tipo de objeto y no todas las acciones se pueden realizar con todos los objetos, por ejemplo, no se puede martillar con un cuchillo o cortar con un martillo. Las interacciones posibles de la base de datos, se muestran en la Tabla 4.1, donde por ejemplo, en la primera fila se tiene que el objeto definido como (Pelota) presenta tres ofrecimientos, es decir que, dada la base de datos, solo se realizan tres acciones (Agarrar, Levantar o Empujar). Por último, se resalta que las diferentes interacciones fueron realizadas por 105 maestros humanos diferentes, pero algunos de ellos realizaron las acciones más de una vez. Posteriormente la base de datos se dividió en función a los sujetos, de este modo el conjunto de entrenamiento quedó compuesto por 98 sujetos que realizan 5255 interacciones y el conjunto de validación quedó compuesto por 82 sujetos que realizan 4428 interacciones. Aquí se debe tener en cuenta que para este proyecto se tomaron los conjuntos de entrenamiento y validación de la misma forma en que se seleccionaron por Thermos.

Tabla 4.1 Relaciones Objeto-Acción de la base de datos

Objtos	Acciones												
	Cortar	Agarrar	Martillar	Levantar	Abrir	Pintar	Verter	Empujar	Rotar	Exprimir	Digitar	Desbloquear	Escribir
Pelota		X		X				X					
Libro		X	X	X	X			X					
Botella		X		X			X	X					
Caja		X		X	X			X	X				
Brocha		X		X		X							
Lata		X		X				X					
Pocillo		X		X				X	X				
Martillo		X	X	X									
Llave	X	X		X								X	
Cuchillo	X	X		X									
Boligrafo		X		X									X
Jarra		X		X			X	X	X				
Telefono		X		X				X			X		
Esponja		X		X				X	X	X			

Un aspecto importante a tener en cuenta es que para este proyecto se adicionaron 7 efectos, planteados y seleccionados sólo para el desarrollo de este proyecto, los cuales se reparten entre las 54 interacciones. Esta adición se realiza con la idea de que más adelante se utilice el modelo de los ofrecimientos aprendidos en tareas de inferencia y manipulación robótica, donde toma un mayor interés el conocimiento de los efectos producidos al ejecutar una acción sobre un objeto.

Los 7 efectos adicionados se enlistan a continuación, donde, inicialmente se tiene una sigla acompañada de una breve descripción de su significado. El movimiento en el eje Z indica una elevación del objeto, el eje X es positivo hacia el enfrente de quien realiza la acción y el eje Y es positivo hacia la derecha de quien realiza la acción.

Efectos

1. E0 (Estacionario, sin movimiento)
2. TZ (Transitorio, movimiento en el eje Z)
3. TZXY (Transitorio, movimiento en el eje Z + movimiento en los ejes X y Y)
4. TZG (Transitorio, movimiento en el eje Z + giro)
5. PX (Permanente, cambio en el eje X)
6. PG (Permanente, giro)
7. PM (Permanente, modificación del área del objeto)

En la Tabla 4.2 se presenta cómo se combinan los 7 efectos planteados con las 54 relaciones de interacción existentes.

Tabla 4.2 Interacciones Acción-Objeto-Efecto

Interacción	Acción	Objeto	Efecto	Interacción	Acción	Objeto	Efecto
1	Cortar	Cuchillo	TZXY	**28**	Levantar	Cuchillo	TZ
2	Cortar	Llave	TZXY	**29**	Levantar	Bolígrafo	TZ
3	Agarrar	Pelota	E0	**30**	Levantar	Jarra	TZ
4	Agarrar	Libro	TZ	**31**	Levantar	Teléfono	TZ
5	Agarrar	Botella	E0	**32**	Levantar	Esponja	TZ
6	Agarrar	Caja	E0	**33**	Abrir	Libro	PM
7	Agarrar	Brocha	TZ	**34**	Abrir	Caja	PM
8	Agarrar	Lata	E0	**35**	Pintar	Brocha	TZXY
9	Agarrar	Pocillo	E0	**36**	Verter	Botella	TZG
10	Agarrar	Martillo	TZ	**37**	Verter	Jarra	TZG
11	Agarrar	Llave	TZ	**38**	Empujar	Pelota	PZ
12	Agarrar	Cuchillo	TZ	**39**	Empujar	Libro	PZ
13	Agarrar	Bolígrafo	TZ	**40**	Empujar	Botella	PZ
14	Agarrar	Jarra	E0	**41**	Empujar	Caja	PZ
15	Agarrar	Teléfono	TZ	**42**	Empujar	Lata	PZ
16	Agarrar	Esponja	E0	**43**	Empujar	Pocillo	PZ
17	Martillar	Libro	TZ	**44**	Empujar	Jarra	PZ
18	Martillar	Martillo	TZ	**45**	Empujar	Teléfono	PZ
19	Levantar	Pelota	TZ	**46**	Empujar	Esponja	PZ
20	Levantar	Libro	TZ	**47**	Girar	Caja	PG
21	Levantar	Botella	TZ	**48**	Girar	Pocillo	PG
22	Levantar	Caja	TZ	**49**	Girar	Jarra	PG
23	Levantar	Brocha	TZ	**50**	Girar	Esponja	PG
24	Levantar	Lata	TZ	**51**	Exprimir	Esponja	TZG
25	Levantar	Pocillo	TZ	**52**	Digitar	Teléfono	E0
26	Levantar	Martillo	TZ	**53**	Desbloquear	Llave	TZG
27	Levantar	Llave	TZ	**54**	Escribir	Bolígrafo	TZG

4.2 Etapa de preprocesamiento

La primera etapa de la metodología propuesta es el preprocesamiento, que está compuesto a su vez de 3 sub-etapas como se observa en la Figura 4.3. Primero se tiene la segmentación, de la cual se obtendrá la imagen sin fondo, que será la entrada al reconocimiento de objetos, adicionalmente se obtienen las imágenes totalmente segmentadas de la mano y el objeto, que serán la entrada al cálculo del flujo óptico. Segundo para el flujo óptico de la mano se consigna en una sola imagen la información

de interés para el reconocimiento de la acción realizada. Tercero para el flujo óptico del objeto, de modo similar al paso anterior se presenta en una sola imagen la información que permite reconocer los efectos en las diferentes interacciones. Más adelante se encontrará información detallada de cada sub-etapa.

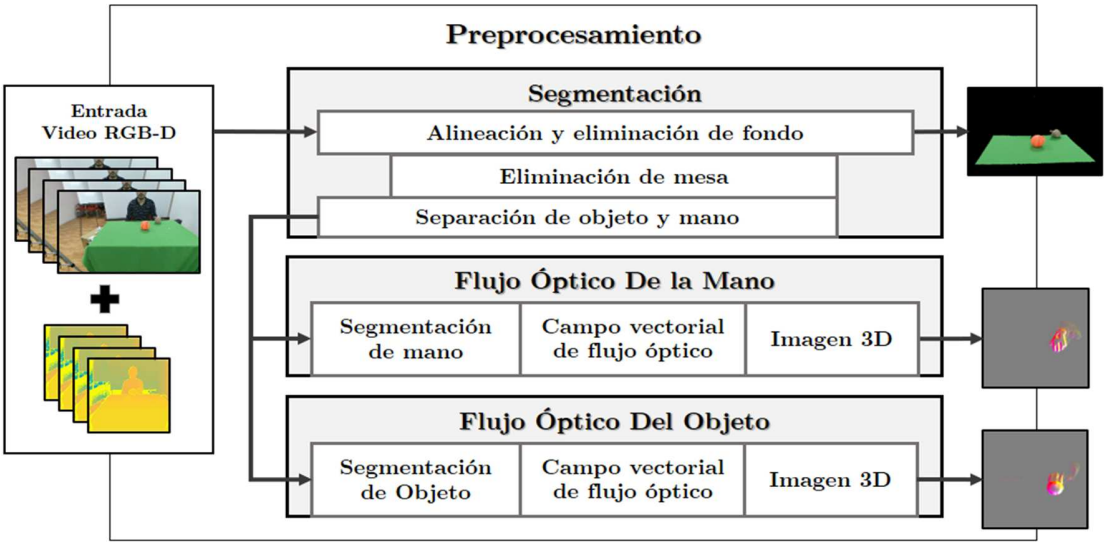


Figura 4.3 Etapa de Preprocesamiento, donde con una segmentación y con dos flujos ópticos se obtienen 3 imágenes.

4.2.1 Segmentación de imagen

El proceso de segmentación planteado está dividido en tres partes, como se observa en la Figura 4.4. Se inicia con los fotogramas de los videos RGB de (1920x1080) y los mapas de profundidad de (512x424). En este punto se ejecuta la alineación y eliminación de fondo, luego se realiza la eliminación de la mesa y por último se lleva a cabo la separación de la mano y el objeto presentes en cada imagen.

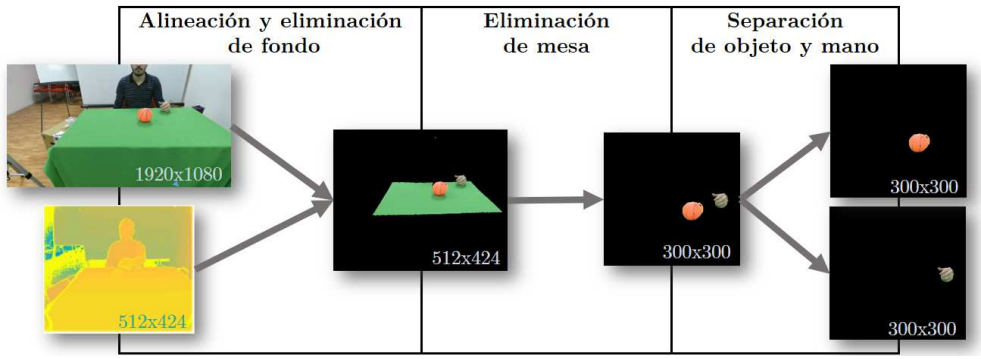


Figura 4.4 Fases del proceso de segmentación, compuesta tres etapas para obtener la segmentación de la mano y el objetos.

Alineación y eliminación de fondo

El proceso de alineación es el primer paso de la segmentación, en éste se ejecuta el código de Matlab, proporcionado por los autores de la base de datos, para alinear las imágenes RGB con las de profundidad y recortar el volumen 3D de interés. La eliminación del fondo se realiza por segmentación basada en umbrales, teniendo en cuenta que los videos fueron tomados bajo condiciones controladas, por lo tanto, el dispositivo de captura siempre se encuentra a la misma distancia y orientación de la región de interés (la superficie plana de la mesa, incluyendo lo que se encuentra encima de ella), así se determinaron previamente los umbrales de la posición en el mundo donde se encuentra la región de interés y se elimina todo lo que está por fuera de la misma.

Eliminación del verde de la mesa

Este segundo proceso de la segmentación consiste en eliminar los pixeles verdes de la imagen que representan la superficie de la mesa cubierta por el mantel, esto se logra también mediante segmentación de color basada en umbrales. En este caso se ejecuta el código de C++ del repositorio github entregado igualmente por los autores de la base de datos, en este se eliminan los pixeles que no se encuentren dentro de los umbrales previamente definidos (en el espacio RGB, la componente G entre 50 - 215 y del espacio HSV, las componentes H entre 5 – 75, S mayor a 50 y V entre 25 - 215).

Separación de objeto y mano

Como tercer y último proceso de la segmentación se realiza una separación de las componentes de las imágenes, que a esta altura solo tienen los objetos y las manos que realizan las acciones. Esta se efectúa nuevamente basada en segmentación por color, aquí se tiene en cuenta que los pixeles que representan la piel de la mano se encuentran dentro de unos umbrales previamente definidos (en el espacio HSV, las componentes H entre 8 – 58, S menor a 90 y V entre 90 - 220), de este modo se obtiene una imagen que contiene sólo el objeto sobre el que se realiza la acción y en otra imagen la mano que realiza la acción.

4.2.2 Cálculo de flujo óptico

Como se mencionó anteriormente en esta etapa se calculan dos flujos ópticos. El primero, que toma como entrada la secuencia de fotogramas de la mano segmentada y la respuesta que entrega, será la entrada utilizada para hacer el reconocimiento de

las acciones. El segundo, toma la secuencia de fotogramas de los objetos segmentados y nos entrega la imagen que será la entrada a la red que hace el reconocimiento de los efectos.

Después de la segmentación se calculan los flujos ópticos, como se observa en la Figura 4.5. Se inicia con una secuencia de n fotogramas ya segmentados, de los cuales se toma una imagen de por medio, para agilizar el cálculo de los campos vectoriales para todo el video, es decir que se obtiene $(n-1)/2$ campos vectoriales que representan todo el del flujo óptico. Después a cada uno de estos campos se le extrae la magnitud, la componente X y componente Y. Dichas componentes se suman de manera independiente para acumular el flujo de toda la interacción en una sola matriz para cada componente y se normalizan a una escala de 0 a 255 tomando el 127 como valor neutro, ya que las componentes vectoriales pueden ser tanto positivas como negativas. Por último, se genera una sola imagen 3D donde queda consignado el flujo óptico de toda la interacción, donde la primera dimensión almacena los valores de la magnitud, la segunda almacena la componente del flujo en el eje X y la última dimensión almacena los valores de la componente del flujo en el eje Y. Después de que se tienen la segmentación y los flujos ópticos calculados se pasa a la etapa de reconocimiento inicial.

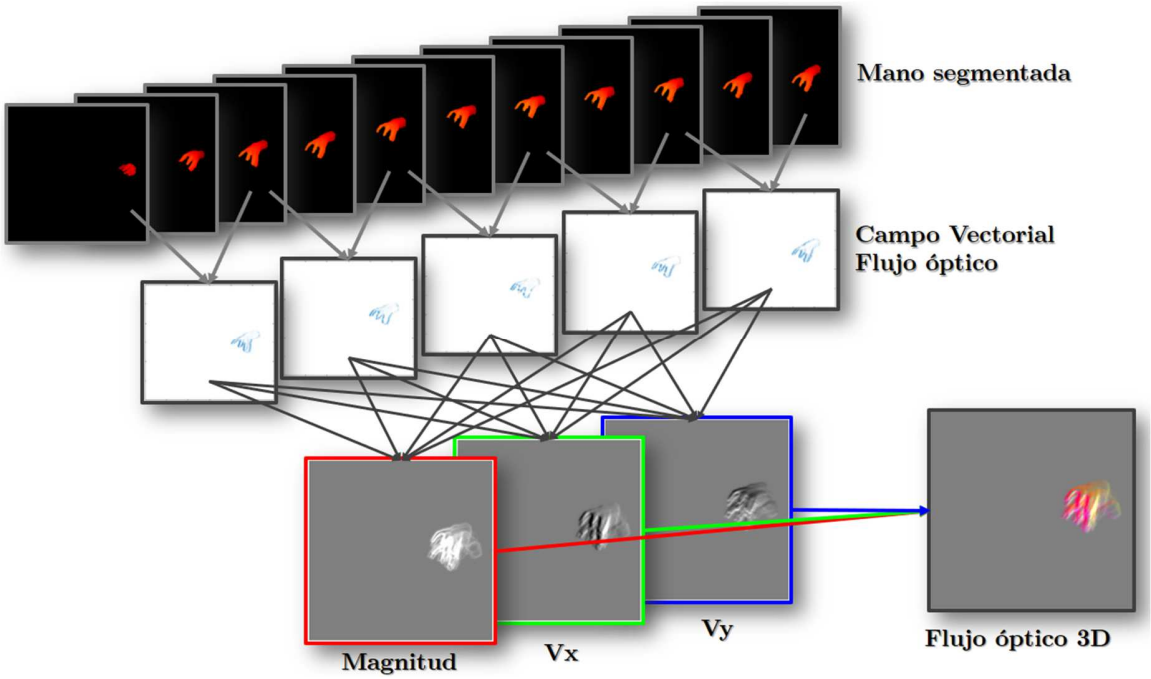


Figura 4.5 Proceso de computación del flujo óptico, donde a partir de un conjunto de fotogramas segmentados se obtiene una imagen de flujo óptico.

4.3 Etapa de reconocimiento inicial

Esta etapa de reconocimiento también se encuentra compuesta de 3 sub-etapas como se observa en la Figura 4.6. Las tres CNNs utilizadas para el reconocimiento inicial son independientes entre sí, ya que no comparten ningún tipo de información. Las redes se alimentan con las imágenes obtenidas en la etapa anterior de preprocesamiento y entregan como salida tres vectores de reconocimiento: uno para los objetos, otro para las acciones y el último para los efectos. Estas estimaciones serán la entrada a la etapa de fusión donde se espera obtener una mejora en el reconocimiento de cada componente.

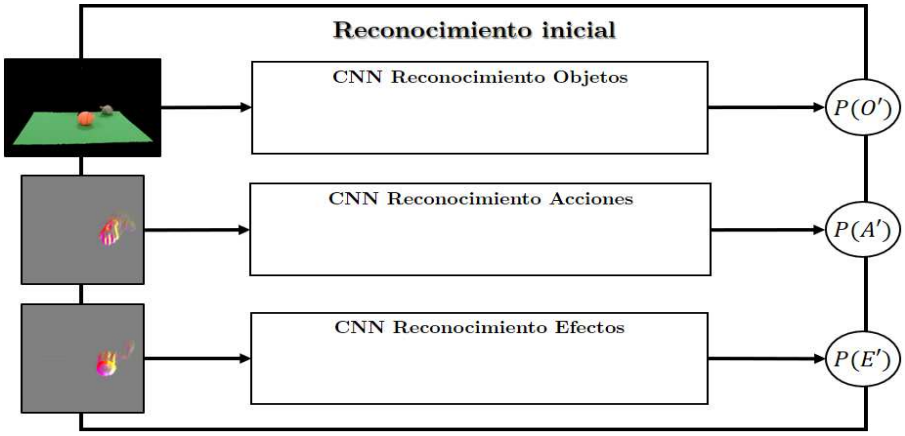


Figura 4.6 Etapa de Reconocimiento inicial

Las tres CNNs son muy similares como se observa en la Figura 4.7, la Figura 4.8 y la Figura 4.9. Todas tienen una arquitectura multicapa similar a la llamada AlexNet (Krizhevsky, et al., 2012). Nuestras redes consisten en alternar 2 capas convolucionales, seguidas por 2 capas completamente conectadas que conducen a un clasificador softmax. En la capa 1 se tiene una convolución con una ventana $(5 \times 5 \times 3)$ y se aplican 64 filtros, después se pasa por una reducción máxima de (2×2) y una normalización. Para la capa 2 se tiene una convolución con una ventana $(5 \times 5 \times 64)$ y se aplican 64 filtros, después se pasa por una normalización y una reducción máxima de (2×2). La capa 3 es completamente conectada y pasa de $(16 \times 16 \times 64)$ a (384) parámetros. La capa 4 también es completamente conectada y pasa de (384) a (192) parámetros. Por último, la capa 5 es un clasificador softmax, donde el número de clases de salida varía en función a la componente de la interacción que se desea reconocer, ya sea uno de los (14) objetos, una de las (13) acciones o uno de los (7) efectos.

La selección de la arquitectura basada en la red AlexNet tiene varios criterios a considerar. Para empezar, ésta ha sido muy utilizada en la literatura. Por otro lado, con esta red se han obtenido buenos resultados en diversos problemas de clasificación de imágenes como en Imagenet. Por último, el criterio más importante para seleccionarla fue la rapidez de entrenamiento ya que sólo tiene 5 capas, con 2 de ellas convolucionales. En cuanto al entrenamiento de cada red se tomaron las 5255 imágenes del conjunto de entrenamiento divididas en lotes aleatorios de 64 imágenes, se realiza con 2000 pasos de entrenamiento, y toma un tiempo de alrededor de 30 minutos en el equipo de cómputo utilizado, el cual se describen la sección 5.1. Los parámetros de entrenamiento enlistados anteriormente son el resultado de la primera parte de la etapa experimental, donde se podrá encontrar de donde se obtuvieron estos valores con más detalle.

En este punto se debe tener en cuenta que, aunque las redes son relativamente sencillas, cumplen con su objetivo que básicamente consiste en obtener una estimación inicial, la cual se utilizará como la entrada a la BN planteada para modelar los ofrecimientos, los cuales constituyen el objetivo principal estudiado en este proyecto.

4.3.1 Reconocimiento inicial de objeto

Esta primera CNN encargada de reconocer inicialmente los objetos, toma como entrada el primer fotograma del video, pero después de la segmentación donde se ha eliminado el fondo. Esta imagen se redimensiona a 64×64 pixeles y se procesa en la red que se observa en la Figura 4.7, obteniendo como salida una lista con las probabilidades de reconocimiento estimado para los 14 objetos disponibles en la base de datos.

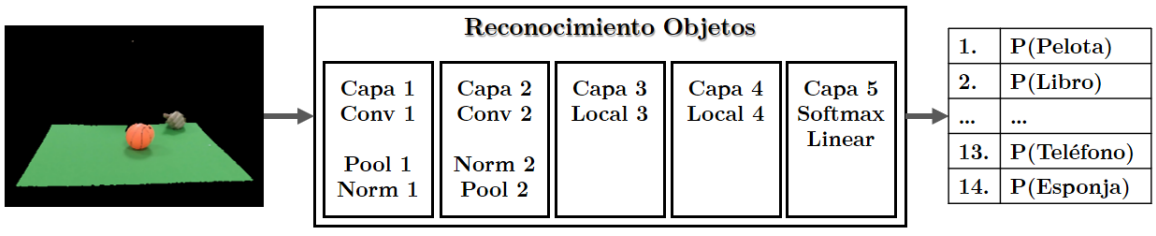


Figura 4.7 Reconocedor de objetos, CNN de 5 capas alimentada por un fotograma sin fondo y entrega un vector de 14 probabilidades.

4.3.2 Reconocimiento inicial de acción

Después se tiene la CNN encargada de reconocer inicialmente las acciones, la cual toma como entrada la imagen 3D del flujo óptico acumulado de la mano que realiza la acción. Ésta también es redimensionada a 64×64 pixeles, para ser procesada por la red que se observa en la Figura 4.8, obteniendo como salida una lista con las probabilidades de reconocimiento estimado para las 13 acciones posibles.

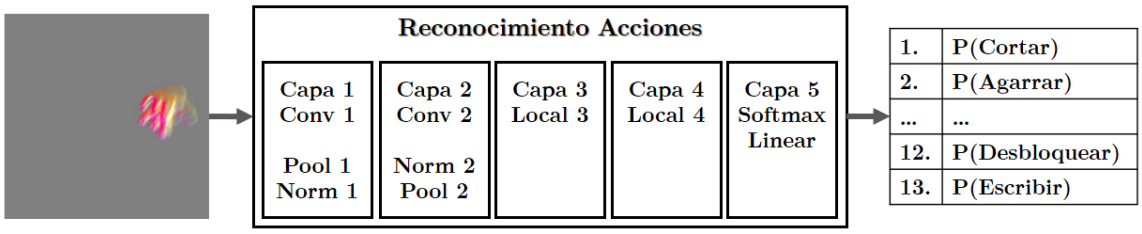


Figura 4.8 Reconocedor de acciones, CNN de 5 capas alimentada por el flujo óptico de la mano y entrega un vector de 13 probabilidades.

4.3.3 Reconocimiento inicial de efecto

Por último, se tiene la CNN de reconocimiento de efectos, la cual utiliza como entrada la imagen 3D del flujo óptico acumulado del objeto sobre el que se realiza la acción, ésta es procesada en la red que se observa en la Figura 4.9 y se obtiene como salida un vector, este con las probabilidades de reconocimiento de los diferentes 7 efectos.

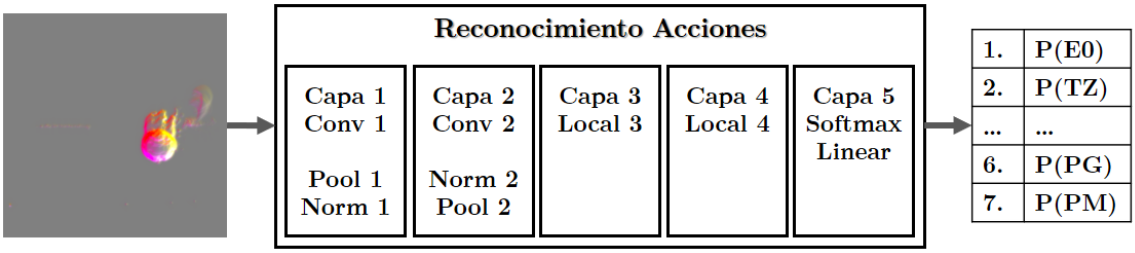


Figura 4.9 Reconocedor de efectos, CNN de 5 capas alimentada por el flujo óptico del objeto y entrega un vector de 14 probabilidades.

4.4 Etapa de fusión

La última etapa de la metodología propuesta consiste en la fusión de las 3 estimaciones iniciales, en esta etapa se modelan los ofrecimientos de los objetos como una red bayesiana multidimensional que relaciona los objetos, acciones y efectos presentes en cada interacción. Por un lado, esta BN tiene tres atributos como

variables de entrada que son: (O') que representa la estimación inicial del objeto presente en la interacción, (A') que representa la estimación inicial de la acción y (E') que representa la estimación inicial del efecto, todas ellas proporcionadas por las CNNs de la etapa anterior. Por el otro lado, la BN tiene tres clases como variables de salida que son: (O) que representa los objetos, (A) que representa las acciones y (E) que representa los efectos. Estas salidas se consideran estimaciones mejoradas, ya que son producto de la inferencia de la BN basada en las tres variables iniciales, aunque se debe tener en cuenta que el modelo también es capaz de lidiar con la incertidumbre, así se puede obtener la inferencia sin importar que falte la información concerniente a alguna de las variables de entrada.

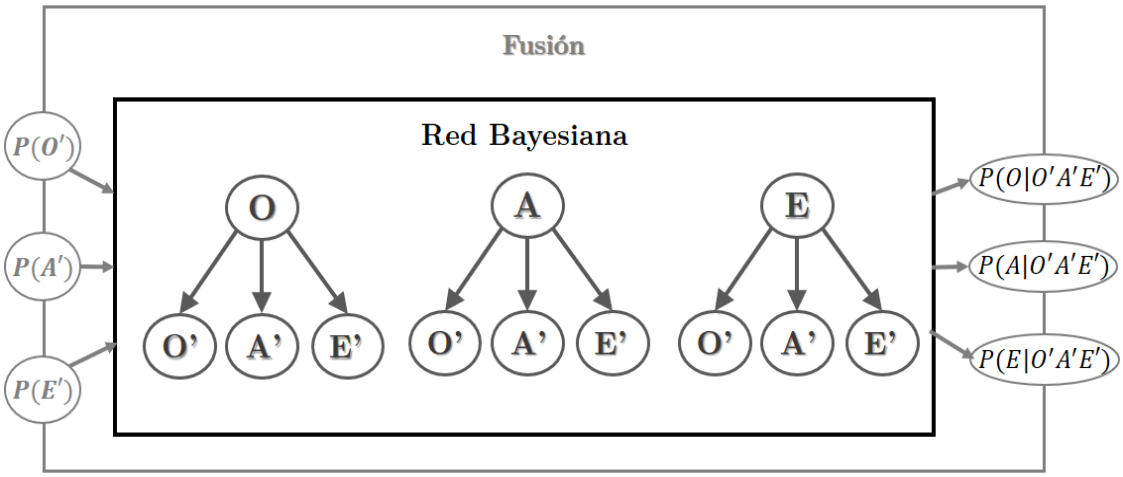


Figura 4.10 Etapa de fusión

La estructura de la red bayesiana multidimensional se definió como se observa en la Figura 4.10 donde se tienen 3 clasificadores bayesianos simples (CBS). Dichos clasificadores se tomaron como CBS ya que de este modo se reduce drásticamente la complejidad del clasificador bayesiano en espacio y tiempo de cálculo (Sucar, 2008). De este modo cada variable de salida se conecta a las 3 variables de entrada que son las estimaciones iniciales de objetos, acciones y efectos, pero las variables de salida no tienen ninguna conexión entre sí. Cada una de las sub-redes tiene asociado un conjunto de cuatro CPTs, donde se almacenan los parámetros que permiten realizar los cálculos de inferencia. A continuación, se presentan las dimensiones que tienen las CPTs requeridas por cada sub-red. Las matrices de probabilidades con sus valores se presentan en el apéndice A, ya que se probaron diferentes valores para estos parámetros.

CPTs de la sub-red de Objetos

P(O) → Vector de [1 x 14]

P(A'|O) → Matriz de [14 x 13]

P(O'|O) → Matriz de [14 x 14]

P(E'|O) → Matriz de [14 x 7]

CPTs de la sub-red de Acciones

P(A) → Vector de [1 x 13]

P(A'|A) → Matriz de [13 x 13]

P(O'|A) → Matriz de [13 x 14]

P(E'|A) → Matriz de [13 x 7]

CPTs de la sub-red de Efectos

P(A) → Vector de [1 x 7]

P(A'|A) → Matriz de [7 x 13]

P(O'|A) → Matriz de [7 x 14]

P(E'|A) → Matriz de [7 x 7]

Como se mencionó anteriormente los CPTs utilizados como parámetros de las BN se calcularon de 4 formas diferentes. Los primeros dos grupos de CPTs se calculan a partir de la distribución de las variables en la definición de la base de datos, estos llevaran los nombres de **CPTs *uniformes*** y **CPTs *uniformes suavizados***. Los otros dos grupos de CPTs se calcularon en función a las estimaciones iniciales de los ejemplos de entrenamiento de la base de datos, estos llevaran los nombres de **CPTs *estimaciones suaves*** y **CPTs estimaciones *duras***. A continuación, se presentan las 4 diferentes tablas de probabilidad condicional de objeto dada la acción P(A'|O) como ejemplo de las diferentes formas de calcular las CPTs, teniendo en cuenta que las demás tablas que componen las CPTs se consignan en el Anexo B, estas están acompañadas de una pequeña descripción del proceso utilizado para realizar los diferentes cálculos.

4.4.1 CPTs uniformes

Para establecer las CPTs requeridas como parámetros de la BN que modela los ofrecimientos de los objetos, inicialmente, nos basamos solo en la distribución de las variables de la base de datos, estos se puede observar en la Tabla 4.1 donde se presentan las relaciones entre las variables objetos y acciones de las 54 posibles interacciones. Para el cálculo de la CPT se normaliza cada fila haciendo que la sumatoria de las posibles interacciones sea igual a 1, de este modo se obtiene la P(A'|O) suponiendo una distribución uniforme de las acciones para cada objeto con el que se relaciona. De este modo el objeto (Pelota) presenta una probabilidad 0.33 de ser agarrada, 0.33 de ser levantada y 0.33 de ser empujada, como se observa en la primera fila de la Tabla 4.3. Por otro lado, las demás acciones, como cortar o martillar entre otras, que según la base de datos no se pueden realizar con el objeto pelota tendrán una probabilidad de 0.

Tabla 4.3 CPT de P(A'|O) obtenida de la distribución uniforme de la base de datos

P(A|O)		A1	A2	A3	A4	A5	A6	A7	A8	A9	A10	A11	A12	A13
		Cortar	Agarrar	Martillar	Levantar	Abrir	Pintar	Verter	Empujar	Girar	Exprimir	Digitar	Desbloquear	Escribir
O1	Pelota	0	0.33	0	0.33	0	0	0	0.33	0	0	0	0	0
O2	Libro	0	0.20	0.2	0.20	0.2	0	0	0.20	0	0	0	0	0
O3	Botella	0	0.25	0	0.25	0	0	0.25	0.25	0	0	0	0	0
O4	Caja	0	0.20	0	0.20	0.2	0	0	0.20	0.2	0	0	0	0
O5	Brocha	0	0.33	0	0.33	0	0.33	0	0	0	0	0	0	0
O6	Lata	0	0.33	0	0.33	0	0	0	0.33	0	0	0	0	0
O7	Pocillo	0	0.25	0	0.25	0	0	0	0.25	0.25	0	0	0	0
O8	Martillo	0	0.33	0.3333	0.33	0	0	0	0	0	0	0	0	0
O9	Llave	0.25	0.25	0	0.25	0	0	0	0	0	0	0	0.25	0
O10	Cuchillo	0.33	0.33	0	0.33	0	0	0	0	0	0	0	0	0
O11	Bolígrafo	0	0.33	0	0.33	0	0	0	0	0	0	0	0	0.333
O12	Jarra	0	0.20	0	0.20	0	0	0.2	0.20	0.2	0	0	0	0
O13	Teléfono	0	0.25	0	0.25	0	0	0	0.25	0	0	0.25	0	0
O14	Esponja	0	0.20	0	0.20	0	0	0	0.20	0.2	0.2	0	0	0

4.4.2 CPTs uniformes suavizados

Después se realizaron pruebas con un conjunto de CPTs muy similares a las anteriores, con la diferencia de que esta vez las distribuciones uniformes obtenidas anteriormente se suavizaron cambiando los valores de las probabilidades iguales a 0 por un valor de 0.01 y normalizando nuevamente para que la sumatoria de las probabilidades de cada fila sea igual a 1, de este modo se obtuvieron las CPTs que se observan en la Tabla 4.4. esta modificación de realizo con el ánimo de que el sistema pueda lidiar con errores en las estimaciones de entrada.

Tabla 4.4 CPT de la P(A'|O) obtenida de la distribución uniforme suvizada de la base de datos.

P(A|O)		A1 Cortar	A2 Agarrar	A3 Martillar	A4 Levantar	A5 Abrir	A6 Pintar	A7 Verter	A8 Empujar	A9 Girar	A10 Exprimir	A11 Digitar	A12 Desbloquear	A13 Escribir
O1	Pelota	0.01	0.30	0.01	0.30	0.01	0.01	0.01	0.30	0.01	0.01	0.01	0.01	0.01
O2	Libro	0.01	0.18	0.184	0.18	0.18	0.01	0.01	0.18	0.01	0.01	0.01	0.01	0.01
O3	Botella	0.01	0.23	0.01	0.23	0.01	0.01	0.23	0.23	0.01	0.01	0.01	0.01	0.01
O4	Caja	0.01	0.18	0.01	0.18	0.18	0.01	0.01	0.18	0.18	0.01	0.01	0.01	0.01
O5	Brocha	0.01	0.30	0.01	0.30	0.01	0.3	0.01	0.01	0.01	0.01	0.01	0.01	0.01
O6	Lata	0.01	0.30	0.01	0.30	0.01	0.01	0.01	0.30	0.01	0.01	0.01	0.01	0.01
O7	Pocillo	0.01	0.23	0.01	0.23	0.01	0.01	0.01	0.23	0.23	0.01	0.01	0.01	0.01
O8	Martillo	0.01	0.30	0.3	0.30	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
O9	Llave	0.23	0.23	0.01	0.23	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.23	0.01
O10	Cuchillo	0.3	0.30	0.01	0.30	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
O11	Bolígrafo	0.01	0.30	0.01	0.30	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.3
O12	Jarra	0.01	0.18	0.01	0.18	0.01	0.01	0.18	0.18	0.18	0.01	0.01	0.01	0.01
O13	Teléfono	0.01	0.23	0.01	0.23	0.01	0.01	0.01	0.23	0.01	0.01	0.228	0.01	0.01
O14	Esponja	0.01	0.18	0.01	0.18	0.01	0.01	0.01	0.18	0.18	0.184	0.01	0.01	0.01

4.4.3 CPTs estimaciones suaves

Los CPTs obtenidos hasta el momento demostraron ser útiles ya que ayudaban a la BN a cumplir su tarea de mejorar las estimaciones de reconocimiento de objetos acciones y efectos. Pero decidimos calcular otros dos grupos de CPTs, los cuales no se limitan sólo a la distribución de las variables en la base de datos, sino que se tomaron los ejemplos etiquetados del conjunto de entrenamiento. Este enfoque de aprender los parámetros de los ejemplos de entrenamiento se adoptó con la idea de que se obtendrían CPTs mejores a las calculadas de la distribución de la base de datos, ya que los CPTs aprendidos toman en cuenta los errores de las estimaciones de las variables de entrada. En este caso, por ejemplo, si en una interacción se tiene la etiqueta de objeto igual a (pelota) y se observan las estimaciones de las posibles acciones tendríamos un vector de 13 probabilidades como el que se observa a continuación.

P(A|Pelota)		A1 Cortar	A2 Agarrar	A3 Martillar	A4 Levantar	A5 Abrir	A6 Pintar	A7 Verter	A8 Empujar	A9 Girar	A10 Exprimir	A11 Digitar	A12 Desbloquear	A13 Escribir
O1	Pelota	0.000	0.824	0.000	0.156	0.008	0.000	0.000	0.000	0.002	0.000	0.000	0.005	0.000

De este modo, se toman cada uno de los ejemplos del conjunto de entrenamiento y dependiendo de la etiqueta del objeto que se tenga en la interacción, se van sumandos los vectores de las posibles acciones con la respectiva fila que le corresponde. Después se normaliza cada fila para que la suma de todo sus elementos sea igual a 1 y así se obtienen las CTP que llamamos ***estimaciones suaves***, como la que se presenta en

la Tabla 4.5. Estas CPTs se llaman estimaciones suaves por porque tomas los vectores de estimación con todos los valores de probabilidad entregados por las CNNs.

Tabla 4.5 CPT de la P(A'|O) obtenida de los datos de entrenamiento, tomando las *estimaciones suaves.*

P(A|O)		A1 Cortar	A2 Agarrar	A3 Martillar	A4 Levantar	A5 Abrir	A6 Pintar	A7 Verter	A8 Empujar	A9 Girar	A10 Exprimir	A11 Digitar	A12 Desbloquear	A13 Escribir
O1	Pelota	0.01	0.24	0.02	0.25	0.07	0.01	0.02	0.26	0.02	0.02	0.04	0.02	0.01
O2	Libro	0.03	0.17	0.16	0.16	0.18	0.01	0.01	0.20	0.02	0.01	0.01	0.01	0.01
O3	Botella	0.02	0.23	0.01	0.28	0.01	0.01	0.09	0.23	0.03	0.02	0.01	0.02	0.01
O4	Caja	0.02	0.26	0.02	0.19	0.11	0.01	0.02	0.15	0.17	0.01	0.01	0.01	0.01
O5	Brocha	0.01	0.29	0.03	0.30	0.01	0.25	0.01	0.02	0.02	0.01	0.01	0.01	0.01
O6	Lata	0.02	0.28	0.01	0.28	0.01	0.01	0.01	0.29	0.03	0.01	0.01	0.01	0.01
O7	Pocillo	0.02	0.23	0.01	0.24	0.01	0.01	0.02	0.20	0.20	0.01	0.02	0.01	0.01
O8	Martillo	0.01	0.28	0.26	0.28	0.02	0.04	0.01	0.03	0.01	0.01	0.01	0.01	0.02
O9	Llave	0.22	0.19	0.02	0.24	0.01	0.01	0.02	0.06	0.03	0.02	0.02	0.11	0.04
O10	Cuchillo	0.17	0.28	0.02	0.28	0.01	0.01	0.02	0.06	0.02	0.02	0.02	0.04	0.04
O11	Bolígrafo	0.12	0.27	0.02	0.26	0.02	0.01	0.02	0.05	0.02	0.01	0.03	0.04	0.12
O12	Jarra	0.01	0.16	0.01	0.26	0.02	0.01	0.14	0.17	0.14	0.02	0.01	0.02	0.01
O13	Teléfono	0.05	0.24	0.01	0.20	0.01	0.01	0.01	0.28	0.03	0.01	0.11	0.01	0.02
O14	Esponja	0.02	0.22	0.02	0.23	0.03	0.01	0.02	0.20	0.10	0.09	0.03	0.02	0.02

4.4.4 CPTs estimaciones duras

Por último, se probó otra forma de calcular los CPTs a partir de los datos de entrenamiento. Esta es muy similar a la anterior, pero en este caso no se toman directamente los vectores de estimaciones suaves, sino que se convierten a vectores de ***estimaciones duras***. Estas llamadas estimaciones duras se obtienen bancarizando las estimaciones suaves, es decir que se le asigna un valor igual a 1 a la probabilidad mayor y a los demás se les asigna un valor igual a 0. por ejemplo, para el caso mencionado anteriormente donde se tenía la etiqueta de objeto igual a (pelota), la estimación de la acción con mayor probabilidad está dada para la acción agarrar, por lo tanto, se le asigna el valor igual a 1 y a los demás se les asigna el valor igual a 0 como se observa a continuación.

P(A|Pelota)		A1 Cortar	A2 Agarrar	A3 Martillar	A4 Levantar	A5 Abrir	A6 Pintar	A7 Verter	A8 Empujar	A9 Girar	A10 Exprimir	A11 Digitar	A12 Desbloquear	A13 Escribir
O1	Pelota	0	1	0	0	0	0	0	0	0	0	0	0	0

Éstos vectores de estimaciones duras también se van sumando con la respectiva fila que le corresponde a cada ejemplo de entrenamiento y al final se normaliza para que la suma de los elementos de cada fila sea igual a 1. De este modo se obtiene una CPT como la de la Tabla 4.6.

Tabla 4.6 CPT de la P(A'|O) obtenida de los datos de entrenamiento, tomando las *estimaciones duras.*

P(A|O)		A1 Cortar	A2 Agarrar	A3 Martillar	A4 Levantar	A5 Abrir	A6 Pintar	A7 Verter	A8 Empujar	A9 Girar	A10 Exprimir	A11 Digitar	A12 Desbloquear	A13 Escribir
O1	Pelota	0.01	0.24	0.01	0.25	0.07	0.01	0.01	0.27	0.02	0.01	0.03	0.02	0.01
O2	Libro	0.03	0.17	0.16	0.16	0.19	0.01	0.01	0.20	0.02	0.01	0.01	0.01	0.01
O3	Botella	0.02	0.24	0.01	0.28	0.01	0.01	0.10	0.23	0.03	0.02	0.01	0.02	0.01
O4	Caja	0.02	0.26	0.01	0.18	0.11	0.01	0.02	0.15	0.18	0.01	0.02	0.01	0.01
O5	Brocha	0.01	0.29	0.02	0.32	0.01	0.26	0.01	0.01	0.01	0.01	0.01	0.01	0.01
O6	Lata	0.01	0.28	0.01	0.28	0.01	0.01	0.01	0.30	0.02	0.01	0.01	0.01	0.01
O7	Pocillo	0.02	0.22	0.01	0.24	0.01	0.01	0.01	0.21	0.20	0.01	0.02	0.01	0.01
O8	Martillo	0.01	0.27	0.27	0.29	0.01	0.03	0.01	0.03	0.01	0.01	0.01	0.01	0.02
O9	Llave	0.24	0.18	0.02	0.24	0.01	0.01	0.02	0.05	0.04	0.02	0.03	0.11	0.04
O10	Cuchillo	0.18	0.28	0.02	0.29	0.01	0.01	0.01	0.05	0.01	0.02	0.01	0.04	0.04
O11	Bolígrafo	0.12	0.29	0.02	0.27	0.02	0.01	0.01	0.05	0.01	0.01	0.02	0.03	0.12
O12	Jarra	0.01	0.17	0.01	0.26	0.02	0.01	0.14	0.17	0.14	0.02	0.01	0.02	0.01
O13	Teléfono	0.04	0.26	0.01	0.19	0.01	0.01	0.01	0.28	0.02	0.01	0.11	0.01	0.02
O14	Esponja	0.02	0.22	0.02	0.22	0.03	0.01	0.02	0.20	0.10	0.09	0.02	0.02	0.02

4.5 Contribuciones

La metodología presentada tiene varias contribuciones por resaltar. Para empezar la información visual de los videos RGB-D se descompone en tres partes obteniendo, por un lado, una imagen segmentada la cual almacena la información para el reconocimiento de los objetos, luego se tiene el flujo óptico de la mano, que contiene la información para realizar el reconocimiento de las acciones y por último se tiene el flujo óptico del objeto con la información para reconocer los efectos generados. Esta forma de descomponer la información visual de cada interacción no se había llevado a cabo anteriormente.

Para la siguiente etapa, que consiste en el reconocimiento inicial de las componentes se tienen tres CNNs, que, aunque son sencillas, éstas nos permiten tener porcentajes de acierto altos, además de entregarnos a la salida una sola variable para los objetos, otra para las acciones y una más para los efectos. Esto nos permite simplificar el problema ya que no tenemos un conjunto de diferentes variables que representan cada componente, sino una sola variable discreta para describir cada componente.

Para terminar, la etapa de fusión que modela los ofrecimientos con una BN permite realizar varias tareas al mismo tiempo ya que además del reconocimiento mejorado de los objetos, las acciones y los efectos, la BN puede lidiar con la información faltante de alguna de las variables y realizar tareas de inferencia útiles en procesos de

manipulación robótica, como selección del objeto a manipular, planeación de acciones a realizar y predicción del efecto causado.

4.6 Resumen

En esta sección se presentó el método propuesto para modela los ofrecimientos de los objetos. Inicialmente, se establece la entrada que serán los videos de la base de datos CERTH-SOR3D. Luego se establece la etapa de preprocesamiento compuesta de una fase de segmentación de imagen y el cálculo de flujo óptico. Después, se tiene la etapa de reconocimiento inicial de los objetos, acciones y efectos, dado por un conjunto de 3 CNNs. Por último, se tiene etapa de fusión que está dada por una BN que tendrá diferentes CPTs como parámetros para llevar a cabo las tereas de mejora en el reconocimiento e inferencia.

En el siguiente capítulo se tiene la etapa de experimentación y además se consigna los resultados obtenidos en las diferentes pruebas realizadas. Para empezar, se presentan las características del equipo de cómputo utilizado para llevar a cabo los experimentos, luego se presentan los experimentos que llevaron a la elección de algunos parámetros y se finaliza presentando los resultados en las tareas propuestas de mejorar el reconocimiento e inferencia de variables desconocidas.

Capítulo 5

Experimentación y resultados

La experimentación que se lleva a cabo en el desarrollo de este proyecto se puede dividir en dos partes. En la primera parte se determinan algunos parámetros como, el tamaño del lote y el número de pasos de entrenamiento de las CNNs que realizan los reconocimientos iniciales. Como segunda parte se prueban diferentes parámetros de la BN de la etapa de fusión, para determinar cuál alcanza un mejor desempeño y las ventajas que ofrece. Para finalizar con este capítulo se hace un análisis de los resultados obtenidos y así se determina si la metodología propuesta en este proyecto cumple sus objetivos de mejorar las habilidades de reconocimiento de los objetos, las acciones y los efectos presentes en cada interacción.

5.1 Características del equipo de cómputo

Antes de pasar a presentar los parámetros seleccionados para definir tanto la CNN como la BN y los resultados obtenidos al ejecutar la metodología propuesta, se realiza una descripción básica del equipo de cómputo en el cual se llevaron a cabo los diferentes experimentos. Empezando por el hardware, el equipo utilizado cuenta, con un procesador Intel Core i5-4440 de cuarta generación, además tiene 6 GiB de memoria RAM. En cuanto al software, se tiene instalado el sistema operativo Ubuntu 16.04 LTS de 64bits, las CNNs son ejecutadas en la librería de Google, TensorFlow 1,5 la cual trabaja en conjunto con el programa Python 3 para entrenar y realizar tareas de inferencia, las BNs con los diferentes parámetros son ejecutados en Matlab 2014 utilizando la librería BNT (Bayes Net Toolbox)

5.2 Definición de parámetros de las redes neuronales convolucionales

Las CNNs utilizadas en el proyecto presentan la arquitectura descrita en el capítulo anterior, con cinco capas, las dos primeras convolucionales acompañadas de agrupamiento y normalización, seguido de dos capas totalmente conectadas y por último se tiene un clasificador softmax. Pero todavía hace falta definir algunos parámetros como el tamaño de lote o el número de pasos de entrenamiento.

5.2.1 Selección de tamaño de lote

Para la definición del tamaño de lote se realizó una prueba, en la cual se entrenó una CNN con diferentes tamaños de lote. Se observó en la Figura 5.1.a que el tiempo de entrenamiento es directamente proporcional al tamaño del lote, por lo tanto se desea entrenar con un lote pequeño, pero si el lote es demasiado pequeño la red diverge, es decir que el entrenamiento algunas veces no concluye como se muestra en la Figura 5.1.b. Así que se desea encontrar un tamaño de lote que sea lo suficientemente pequeño para que el tiempo de entrenamiento sea corto, pero que a su vez tenga una alta probabilidad de concluir el entrenamiento satisfactoriamente. En este punto se realizaron 5 entrenamientos de la red para cada tamaño, de lote de este modo se determinó que un tamaño de lote adecuado para el entrenamiento sería de 64 imágenes ya que toma la mitad de tiempo de un lote de 128 y su porcentaje de éxito es del 80%.

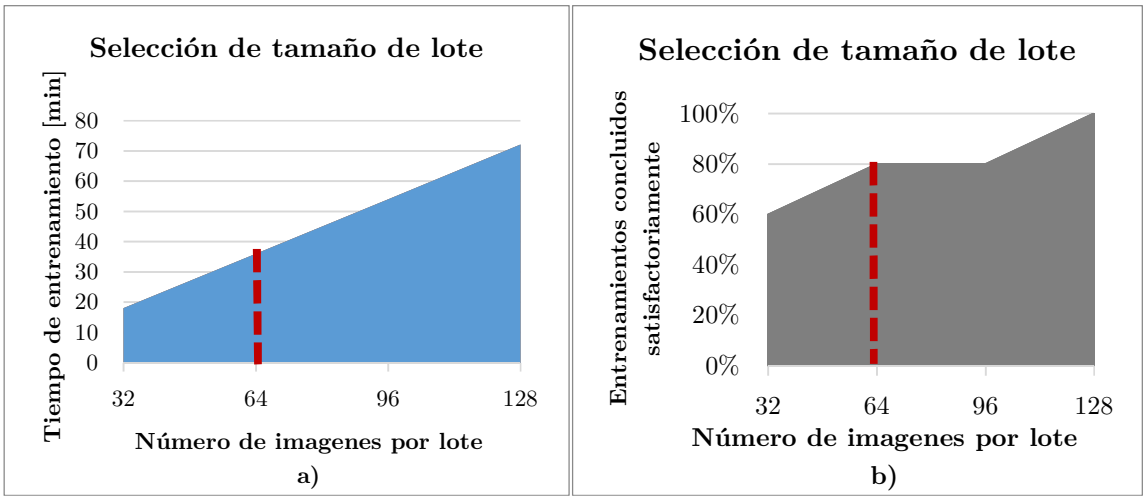


Figura 5.1 Pruebas de tamaño de lote, a) tiempo de entrenamiento vs tamaño de lote, b) entrenamientos concluidos satisgactoriamente vs tamaño de lote.

5.2.2 Pasos de entrenamiento

Para definir el parámetro del número de pasos de entrenamiento se entrenaron varias redes para el reconocimiento de acciones con diferente número de pasos como se observa en la Figura 5.2. Posteriormente se calcula el porcentaje de acierto de cada red, tanto para el conjunto de entrenamiento como para el conjunto de validación. Al analizar la gráfica se puede observar que sin importar cuánto aumente el número de pasos de entrenamiento, el porcentaje de acierto en el conjunto de entrenamiento se estabiliza en un máximo de alrededor del 95%, de forma similar el porcentaje de acierto en el conjunto de validación también llega a un valor en el que se estabiliza alrededor del 75%.

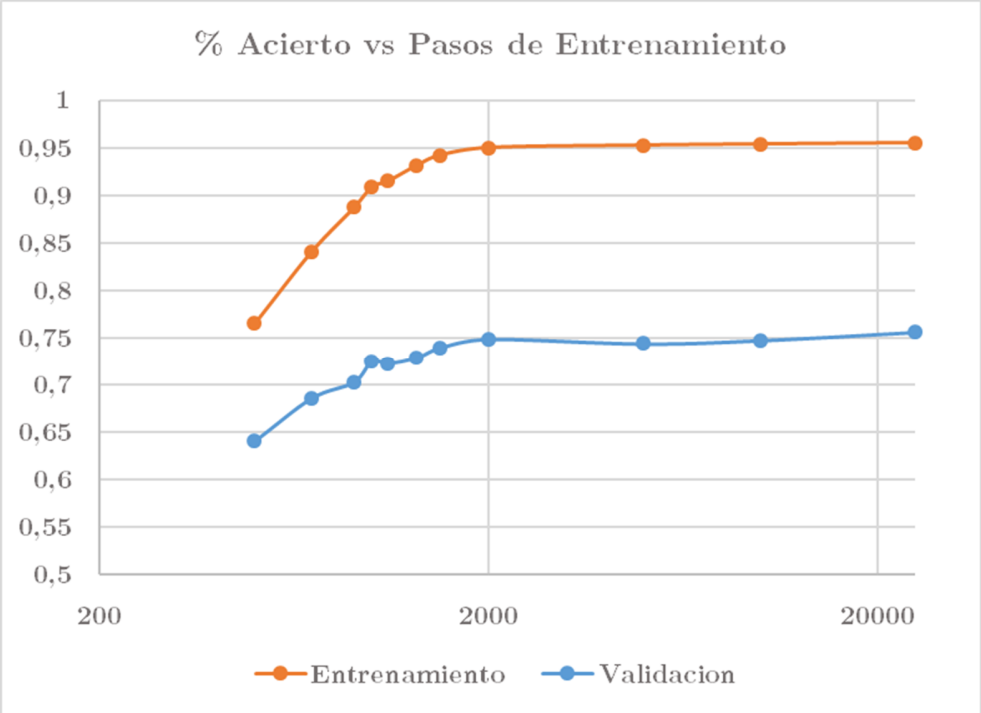


Figura 5.2 Definición de pasos de entrenamiento.

Lo importante a resaltar es que este punto de estabilización se alcanza con solo 2.000 pasos de entrenamiento lo cual permite hacer entrenamientos más rápidos sin pérdida del porcentaje de acierto. Por lo tanto, se define el parámetro de 2.000 pasos de entrenamiento para todas las redes futuras.

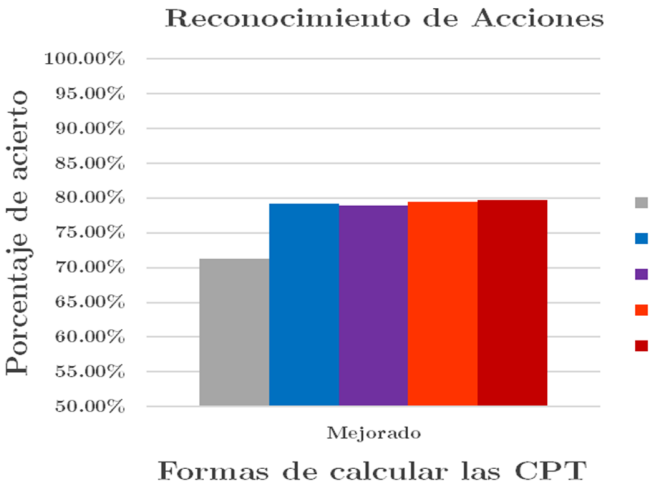
5.3 Prueba de parámetros de la red bayesiana

Esta se puede considerar como la primera etapa experimental donde se pondrá a prueba la metodología propuesta. Esta prueba tiene 2 objetivos específicos; el primero es determinar la importancia de los parámetros de la BN al realizar la tarea de mejora del reconocimiento; el segundo objetivo es establecer cuál de los diferentes métodos presentados en la sección 4.4 para calcular los CPTs de la BN, ofrece un mayor porcentaje de mejora en el reconocimiento de los objetos, las acciones y los efectos. Para el desarrollo de esta prueba, inicialmente se calcula el porcentaje de acierto del reconocimiento inicial dados por las CNNs, con el 100% del conjunto de datos de validación. Posteriormente se calculan los porcentajes de acierto de los reconocimientos mejorados dado por la BN, cuya estructura fue definida también en la sección 4.4 y se le asignan diferentes CPTs, los cuales están consignados en el Anexo B.

En las Figura 5.3, Figura 5.4 y Figura 5.5, las barras de color gris representan el porcentaje de acierto del reconocimiento inicial dado por la CNN. Las barras color azul representan los reconocimientos mejorados al pasar por la BN con ***CPTs uniformes***, que se explicaron en la sección 4.4. Las barras moradas son para la BN con ***CPTs uniformes suavizados***. Las barras de color naranja representan los porcentajes de acierto mejorado, para la BN con ***CPTs estimaciones suaves***. Por último, las barras rojas representan el porcentaje de acierto mejorado para la BN con ***CPTs estimaciones duras***.

En este punto se hace necesario aclarar que los valores reportados en las gráficas son los valores promedio de tres ejecuciones diferentes, las cuales se encuentran consignadas en el Anexo C, donde también se encuentran los valores de las desviaciones estándar.

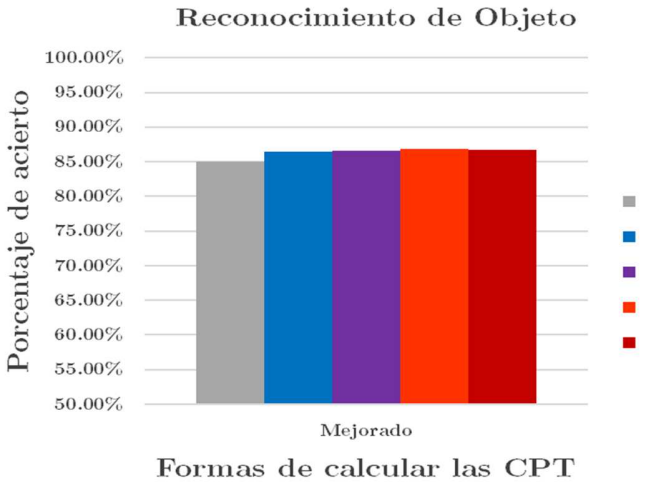
En la gráfica de la Figura 5.3 se puede apreciar que sin importar la forma en que se calculen las CPTs se obtiene una mejora promedio de al menos 7.84%, ésta es una mejora importante la cual es generada con la adición de la información de los objetos y los efectos en el reconocimiento de las acciones. Además, es de resaltar que en este caso la barra roja (que representa los CPTs calculados con estimaciones duras) presenta la mayor mejora llegando a un 8.25% de acierto por encima del reconocimiento inicial.



CPTs	ΔA	Am
Inicial		71.23%
Uniformes	7.62%	79.13%
Uniformes suvizados	7.51%	78.97%
Estimaciones Suaves	8.01%	79.46%
Estimaciones Duras	8.25%	79.71%
Media	7.84%	79.32%

Figura 5.3 Mejora del reconocimiento de acciones para diferentes parámetros de los CPTs de la BN

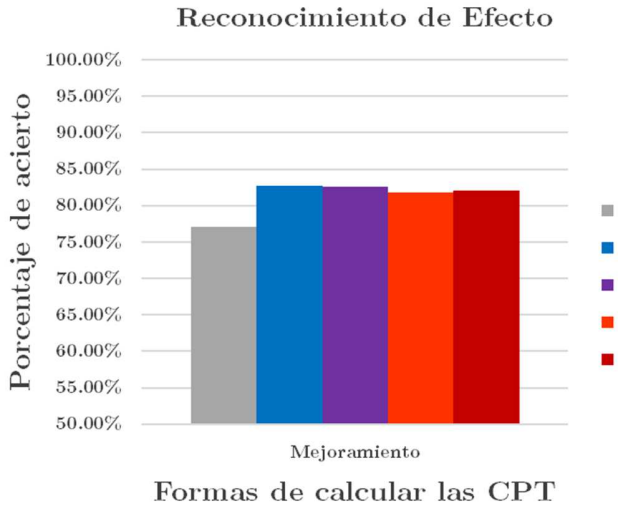
En la gráfica de la Figura 5.4 se puede apreciar que al igual que en el caso anterior sin importar la forma en que se calculen las CPTs se obtiene una mejora promedio de al menos 1.54%, esta mejora es generada con la adición de la información de las acciones y los efectos en el reconocimiento de los objetos. Además, es de resaltar que en este caso la barra naranja (que representa los CPTs calculados con estimaciones suaves) presenta la mayor mejora llegando a un 1.71% de acierto por encima del reconocimiento inicial.



CPTs	ΔO	Om
Inicial		85.02%
Uniformes	1.38%	86.50%
Uniformes suvizados	1.50%	86.58%
Estimaciones Suaves	1.71%	86.81%
Estimaciones Duras	1.58%	86.68%
Media	1.54%	86.64%

Figura 5.4 Mejora del reconocimiento de objetos para diferentes parámetros de los CPTs de la BN

Por último en la gráfica de la Figura 5.5 se puede apreciar que nuevamente sin importar la forma en que se calculen las CPTs se obtiene una mejora promedio de al menos 5.22%, esta mejora es generada con la adición de la información de las acciones y los objetos en el reconocimiento de los efectos. Además, es de resaltar que en este caso la barra azul (que representa las CPTs uniformes calculadas de la base de datos) presenta la mayor mejora llegando a un 5.68% de acierto por encima del reconocimiento inicial.



CPTs	ΔE	Em
Inicial		77.04%
Uniformes	5.68%	82.77%
Uniformes suvizados	5.51%	82.58%
Estimaciones Suaves	4.78%	81.83%
Estimaciones Duras	5.01%	82.05%
Media	5.22%	82.31%

Figura 5.5 Mejora del reconocimiento de efectos para diferentes parámetros de los CPTs de la BN

En este primer experimento, se puede apreciar que la forma de calcular los CPTs no tiene un efecto relevante en los resultados, puesto que, con las 4 formas utilizadas ya sea uniforme, uniforme suavizado, estimaciones suaves o estimaciones duras se obtienen mejoras relativamente equivalentes. Teniendo encueta que los primeros grupos de CPTs se calculan directamente de la distribución de las variables en la base de datos.

Adicionalmente se puede apreciar que el mayor porcentaje de mejora se presenta en el reconocimiento de las acciones, esto se debe a que inicialmente tiene la CNN con el reconocimiento más pobre, por lo tanto, la información adicionada por las otras dos variables en la etapa de fusión con la BN se obtiene una mejora importante. Por otro lado, la mejora más pequeña se presenta en el reconocimiento de los objetos, esto se debe a que inicialmente la CNN que reconoce los objetos es la que tiene el

mejor porcentaje de acierto, por lo tanto, las otras dos variables no tienen la capacidad de aportar mucha más información para obtener un mayor porcentaje de mejora.

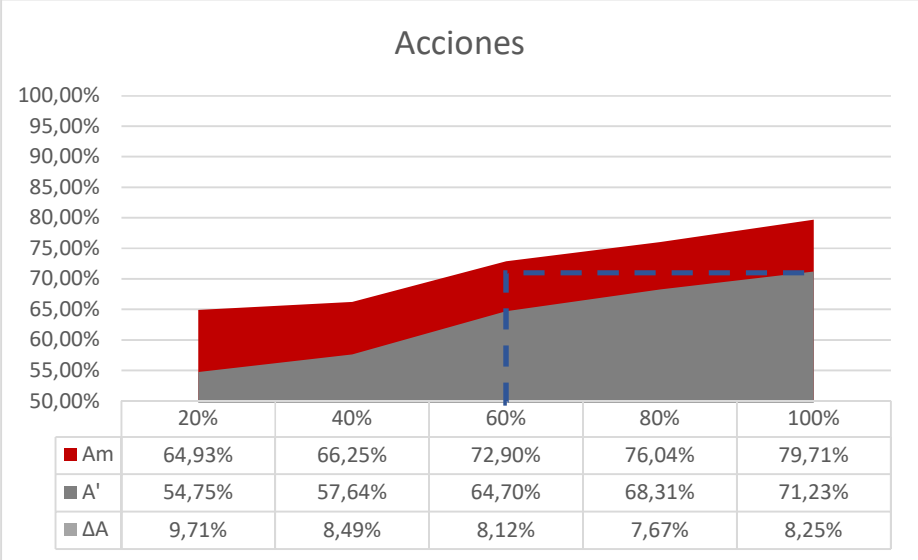
5.4 Conjunto de entrenamiento a utilizar

Este se puede considerar como la segunda prueba de la metodología. Esta prueba tiene 2 objetivos; el primero, es demostrar el comportamiento de la metodología ante CNNs de más bajo porcentaje de reconocimiento; el segundo objetivo planteado era demostrar que se podía llegar al mismo nivel de acierto alcanzado cuando se entrena una CNN con el 100% de los datos de entrenamiento, pero entrenando la CNN con un porcentaje menor de ejemplos y adicionando las mejoras brindadas por la BN que modela los ofrecimientos.

Para esta prueba se fraccionó el conjunto de entrenamiento en 5 partes y se entrenaron 5 tipos de redes diferentes. Para las primeras redes se tomó sólo 1 de las fracciones, que representa el 20% de los datos de entrenamiento, las siguientes redes tomaron 2 de las fracciones que representan el 40%, otras tomaron 3 fracciones que representan el 60%, otras más tomaron 4 fracciones que representan el 80% y las últimas con el 100% de los datos de entrenamiento, cada entrenamiento se realizó 5 veces tomando fracciones diferentes como se describe en el Apéndice A, donde se reportan los valores de acierto para todos los entrenamientos, además del promedio y la desviación estándar. En las gráficas de la Figura 5.6, Figura 5.7 y Figura 5.8 sólo se presentan los valores promedio.

Para el desarrollo de las gráficas, inicialmente se obtiene el porcentaje de acierto inicial entregado por las CNNs de las acciones (A'), de los objetos (O') y de los efectos (E') sobre el conjunto de validación, ésta es la parte gris de las gráficas de la Figura 5.6, Figura 5.7 y Figura 5.8. Posteriormente se calcula el porcentaje de acierto obtenido por la red ya mejorada con la BN utilizando las CPTs estimaciones duras calculadas de los datos de entrenamiento y se grafican en rojo.

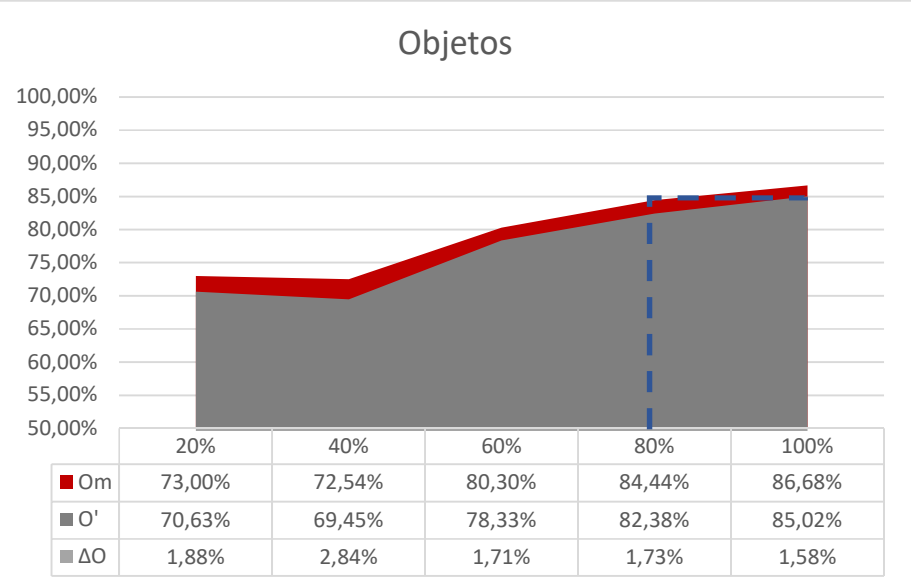
Para empezar, en la Figura 5.6 queda demostrado que para el reconocimiento de las acciones, la parte roja al 60% supera a la parte gris al 100%. Esto indica que la red mejorada (con sólo el 60% de los datos de entrenamiento), tiene mayor porcentaje de acierto que la red sin mejoras (con el 100% de los datos de entrenamiento).



	20%	40%	60%	80%	100%
Am	64,93%	66,25%	72,90%	76,04%	79,71%
A'	54,75%	57,64%	64,70%	68,31%	71,23%
ΔA	9,71%	8,49%	8,12%	7,67%	8,25%

Figura 5.6 Mejoramiento del reconocimiento de acciones, en función al tamaño del conjunto de entrenamiento

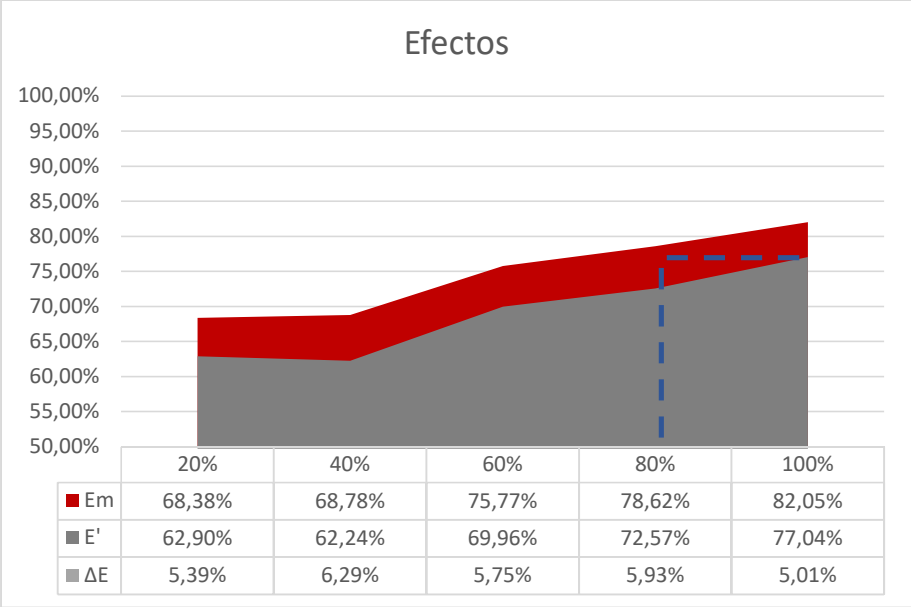
Ahora en la Figura 5.7 se puede ver que para el reconocimiento de los objetos, la parte roja al 80% casi alcanza a la parte gris al 100%. Esto indica que la red mejorada (con solo el 80% de los datos de entrenamiento), tiene casi la misma probabilidad de acierto que la red sin mejoras (con el 100% de los datos de entrenamiento).



	20%	40%	60%	80%	100%
Om	73,00%	72,54%	80,30%	84,44%	86,68%
O'	70,63%	69,45%	78,33%	82,38%	85,02%
ΔO	1,88%	2,84%	1,71%	1,73%	1,58%

Figura 5.7 Mejoramiento del reconocimiento de objetos, en función al tamaño del conjunto de entrenamiento

Por último en la Figura 5.8 se puede ver que para el reconocimiento de los efectos, la parte roja al 80% supera a la parte gris al 100%. Esto indica que la red mejorada (con solo el 80% de los datos de entrenamiento), tiene mayor probabilidad de acierto que la red sin mejoras (con el 100% de los datos de entrenamiento).



	20%	40%	60%	80%	100%
Em	68,38%	68,78%	75,77%	78,62%	82,05%
E'	62,90%	62,24%	69,96%	72,57%	77,04%
ΔE	5,39%	6,29%	5,75%	5,93%	5,01%

Figura 5.8 Mejoramiento del reconocimiento de Efectos, en función al tamaño del conjunto de entrenamiento

En este segundo experimento se puede apreciar que los incrementos del porcentaje de acierto para el reconocimiento de acciones (△A), objetos (△O), y efectos (△E) otorgados por la fusión en la BN, son independientes de la cantidad de ejemplos de entrenamiento de las CNNs. Este resultado es bastante interesante, pues previo al experimento pensamos que el sistema tendería a saturarse, es decir que el porcentaje de mejora al pasar por la BN disminuiría cuando en el reconocimiento inicial de la CNN se tiene un porcentaje de acierto alto.

5.5 Análisis de los resultados

En esta seccion se hace un análisis más detallado de cómo cambian las estimaciones de reconocimiento de las acciones, los objetos y los efectos, entre la etapa de reconocimiento inicial y la etapa de fusión presentada en la metodología. El reconocimiento inicial está dado por las CNNs entrenadas en el caso (a) reportados

en el apéndice B y la fusión está dada por la BN cuyos parámetros de las CPTs se obtuvieron de los datos de entrenamiento, tomando las estimaciones “duras”.

Luego se determinan las matrices de confusión de los reconocimientos iniciales obtenidos por las CNNs y se comparan con las matrices de confusión del reconocimiento mejorado obtenidos por la BN. El objetivo en este punto consiste en determinar explícitamente cuáles son las clases de objetos acciones y efectos, que inicialmente estan ***incorrectamente reconocidos*** (IR) por las CNNs, pero quedan ***correctamente reconocidos*** (CR) al pasar por la BN que realiza la fusión de la informacion. Las matrices de confusión a partir de las cuales se realiza dicho análisis quedan consignadas en el Apéndice D.

5.5.1 Mejoramiento del reconocimiento de acciones

Inicialmente se tiene la Figura 5.9, donde se puede apreciar que la matriz de confusión de las acciones predichas por la BN, presenta mayor cantidad de elementos en su diagonal principal, que la matriz de confusión de las acciones predichas por la CNN lo cual nos indica un mayor número de aciertos en el reconocimiento mejorado.

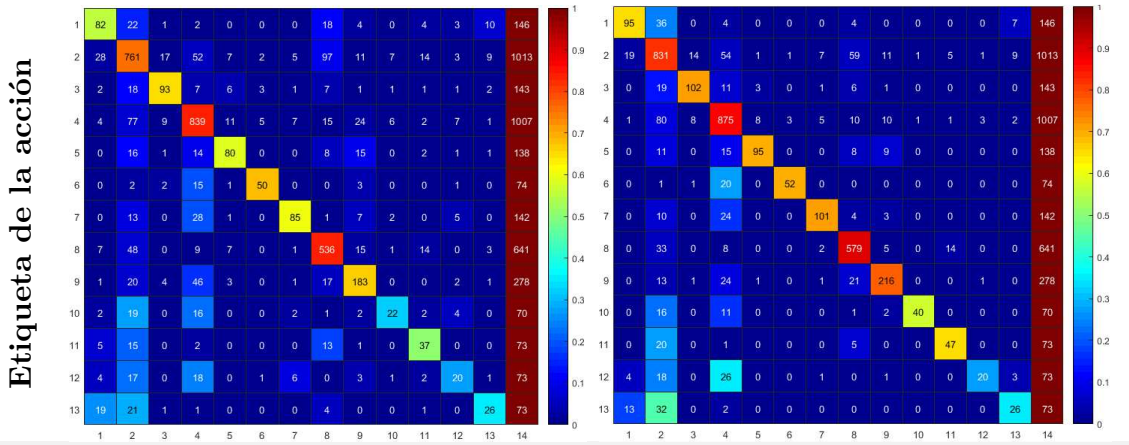


Figura 5.9 Matriz de confusión del reconocimiento de acciones, antes y después de la mejora

Analizando sólo la primera fila de las matrices de la Figura 5.9, se tiene que el reconocimiento inicial de las acciones presenta 82 aciertos, de los 146 casos posibles y el reconocimiento mejorado presenta 95 aciertos, de los 146 casos posibles, aquí se puede decir que el uso de la BN mejora el reconocimiento en 13 casos, sin embargo esta es solo una información parcial.

Para entender un poco mejor lo que está pasando debemos observar la Tabla 5.1, donde se divide la matriz de confusión en cuatro partes, primero se toman los elementos que están CR y al pasar por la red bayesiana siguen estando CR (para el caso que estamos analizando, serían los 78 casos consignados en la fila 1 columna 4 de la Tabla 5.1). Después se toman los elementos que estaban CR al inicio y al pasar por la BN, quedan IR (para el caso que estamos analizando, serían los 4 casos consignados en la fila 1 columna 5 de la Tabla 5.1). Luego se toman los elementos que están IR y al pasar por la BN quedan CR (para el caso que estamos analizando, serían los 17 casos consignados en la fila 1 columna 6 de la Tabla 5.1). Por lo tanto, el cambio de 82 a 95 aciertos de reconocimiento de la acción 1, estaría mejor representado por la operación $82 - 4 + 17 = 95$. Además, en la tabla también se reportan los elementos que estaban IR al inicio y al pasar por la BN siguen estando IR (lo cual no tiene relevancia en el caso que estamos analizando).

Tabla 5.1 Mejoramiento del reconocimiento de acciones

Acciones									
Clases de acciones		**Ejemplos por clase**	**Acción CR al inicio**	**Empieza CR, termina CR**	**Empieza CR, termina IR**	**Empieza IR, termina CR**	**Empieza IR, termina IR**	**Acción CR al final**	
A1	**Cortar**	**146**	**82**	**78**	**4**	**17**	**47**	**↑**	**95**
A2	**Agarrar**	**1013**	**761**	**740**	**21**	**91**	**161**	**↑**	**831**
A3	**Martillar**	**143**	**93**	**89**	**4**	**13**	**37**	**↑**	**102**
A4	**Levantar**	**1007**	**839**	**832**	**7**	**43**	**125**	**↑**	**875**
A5	**Abrir**	**138**	**80**	**76**	**4**	**19**	**39**	**↑**	**95**
A6	**Pintar**	**74**	**50**	**48**	**2**	**4**	**20**	**↑**	**52**
A7	**Verter**	**142**	**85**	**80**	**5**	**21**	**36**	**↑**	**101**
A8	**Empujar**	**641**	**536**	**519**	**17**	**60**	**45**	**↑**	**579**
A9	**Girar**	**278**	**183**	**177**	**6**	**39**	**56**	**↑**	**216**
A10	**Exprimir**	**70**	**22**	**21**	**1**	**19**	**29**	**↑**	**40**
A11	**Digitar**	**73**	**37**	**37**	**0**	**10**	**26**	**↑**	**47**
A12	**Desbloquear**	**73**	**20**	**18**	**2**	**2**	**51**	**-**	**20**
A13	**Escribir**	**73**	**26**	**20**	**6**	**6**	**41**	**-**	**26**
Suma		**3871**	**2814**	**2735**	**79**	**344**	**713**		**3079**
Porcentaje		**100%**	**72.69%**	**70.65%**	**2.04%**	**8.89%**	**18.42%**		**79.54%**

En este punto se debe resaltar que en la Tabla 5.1, los elementos de la columna (Empieza IR, termina CR) son mayores que los de la columna (Empieza CR, termina IR), por lo tanto, para la mayoría de las clases existen mejoras considerables. Pero sin embargo al considerar casos particulares como las acciones ***desbloquear*** o ***escribir*** no se obtiene ninguna mejora en promedio, ya que el número de casos que mejoran son igual al número de casos que empeora. Aunque estas acciones son

exclusivas de un solo sólo objeto, dicho objeto es relativamente pequeño y queda casi completamente ocluido cuando se realiza la acción, este puede ser unos de los motivos para que no mejoren considerablemente.

Adicionalmente, este análisis más fino nos permite observar, en cuáles clases de acciones se tienen las mayores mejoras del reconocimiento antes y después de la BN. En particular la mayor mejora se presenta en la acción ***exprimir*** con un 25,71% de mejora al final de la BN. en este caso la acción también es exclusiva de un solo objetos, pero este es relativamente grande por lo tanto queda ocluido al momento de realizar la acción, este puede ser uno de los motivos para la considerable mejora.

5.5.2 Mejoramiento del reconocimiento de objetos

Por otro lado, en la Figura 5.10 se puede apreciar que la matriz de confusión de los objetos clasificados por la BN también presenta mayor cantidad de elementos en su diagonal principal, que la matriz de confusión de los objetos clasificados por la CNN. Esto nos indica un mayor número de aciertos la pasar por la BN, lo cual indica que el reconocimiento de objetos si mejora.

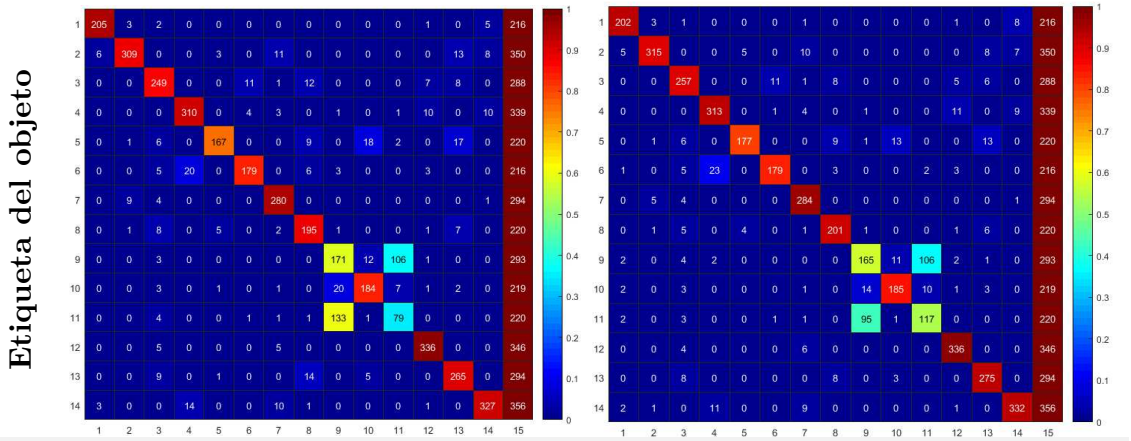


Figura 5.10 Matriz de confusión de reconocimiento de objetos, antes y después de la mejora.

Para el reconocimiento de los objetos, al igual que en el caso anterior los elementos de las diagonales principales de las matrices de confusión de la Figura 5.10 se consignan en las columnas (Objeto CR al inicio) y (Objeto CR al final) de la Tabla 5.2, de este modo podemos determinar para cuáles clases de objetos se obtiene una mejora del reconocimiento, cuáles siguen iguales en promedio y para cuáles empeora el reconocimiento al pasar por la BN.

Tabla 5.2 Mejoramiento del reconocimiento de objetos

Objetos									
Clases de objetos		Ejemplos por clese	Objeto CR al inicio	Empieza CR, termina CR	Empieza CR, termina IR	Empieza IR, termina CR	Empieza IR, termina IR	Objeto CR al final	
O1	Pelota	216	205	202	3	0	11	↓	202
O2	Libro	350	309	309	0	6	35	↑	315
O3	Botella	288	249	249	0	8	31	↑	257
O4	Caja	339	310	310	0	3	26	↑	313
O5	Brocha	220	167	164	3	13	40	↑	177
O6	Lata	216	179	177	2	2	35	-	179
O7	Pocillo	294	280	280	0	4	10	↑	284
O8	Martillo	220	195	195	0	6	19	↑	201
O9	Llave	293	171	92	79	73	49	↓	165
O10	Cuchillo	219	184	182	2	3	32	↑	185
O11	Bolígrafo	220	79	40	39	77	64	↑	117
O12	Jarra	346	336	336	0	0	10	-	336
O13	Teléfono	294	265	265	0	10	19	↑	275
O14	Esponja	356	327	325	2	7	22	↑	332
Suma		3871	3256	3126	130	212	403	3338	
Porcentaje		100%	84.11%	80.75%	3.36%	5.48%	10.41%	86.23%	

En la Tabla 5.2, en general los valores de la columna (Empieza IR, termina CR) son un poco mayores a los valores de la columna (Empieza CR, termina IR), esto nos indica que de forma general se presentan mejoras promedio del 2.12% en el reconocimiento de los objetos. Pero adicionalmente se presentan unos casos relevantes, como en la clase ***bolígrafo*** donde las mejoras obtenidas alcanza en 17.27% de los casos. Por otro lado, se tienen el objeto ***llave*** el cual no mejora, sino que por el contrario presenta el peor caso, con una disminución del reconocimiento del 2.04% al pasar por la BN.

En este caso se observa una confusión entre los objetos bolígrafo y llave que se encuentran en las filas 9 y 11 de la matriz de confusión. Dicha confusión era de esperarse ya que como se mencionó anteriormente estos objetos son relativamente pequeños y se encuentran ocluidos en gran medida cuando se realizan las diferentes interacciones en las que intervienen.

Comparación del reconocimiento de objetos

En este punto se realiza una comparación del mejoramiento en el reconocimiento de los objetos, teniendo en cuenta que ésta es la única tarea en la que nos podemos

comparar con otros autores ya que utilizaron la misma base de datos, la cual fue creada por ellos mismos.

Tabla 5.3 Comparación de las mejoras en el reconocimiento de objetos

Método utilizado	Mejora (%)
Nuestro promedio (CPTs estimaciones suaves)	1.71%
Nuestro mejor caso (CPTs estimaciones duras)	2.12%
Mejor arquitectura GST (Thermos, et al., 2017)	1.38%
Mejor arquitectura GTM (Thermos, et al., 2017)	4.31%

En la Tabla 5.4 inicialmente se presenta la mejora promedio en el reconocimiento de los objetos, obtenida por nuestra metodología cuando la BN utiliza las CPTs calculadas de los datos de entrenamiento con las "estimaciones suaves". Después se presenta la mejora obtenida para nuestro mejor caso el cual se logró cuando la BN utiliza las CPTs calculadas de los datos de entrenamiento con las "estimaciones duras". En tercer lugar se presenta la mayor mejora reportada en (Thermos, et al., 2017) para la arquitectura de red GST presentada en la sección 3.2.1 y por último se presenta la mayor mejora reportada también en (Thermos, et al., 2017) para la arquitectura de red GTM también presentada en la sección 3.2.1.

Se puede apreciar que nuestro porcentaje de mejora es mayor al obtenido por la arquitectura GST, pero inferior al obtenido por la arquitectura GTM. Sin embargo, se debe resaltar la simplicidad de nuestra red (que tiene sólo 2 capas convolucionales, dos totalmente conectadas, un clasificador softmax y una BN. Contra las 13 capas convolucionales más las 3 totalmente conectadas de las VGG-16 utilizadas en cada pilar de las arquitecturas presentadas por Thermos). Por otro lado, se debe resaltar la versatilidad de nuestra metodología ya que permite para realizar otras tareas simultáneamente y no se limita a una sola tarea. Esto hace que nuestro método se muestre como una alternativa interesante.

5.5.3 Mejoramiento del reconocimiento de efectos

La Figura 5.11 resume los resultados para el reconocimiento de efectos, nuevamente se aprecian que la matriz de confusión de los efectos clasificados por la BN, presenta mayor cantidad de elementos en su diagonal principal que la matriz de confusión de los efectos clasificados por la CNN, lo cual nuevamente nos indica un mayor número de aciertos en el reconocimiento mejorado.

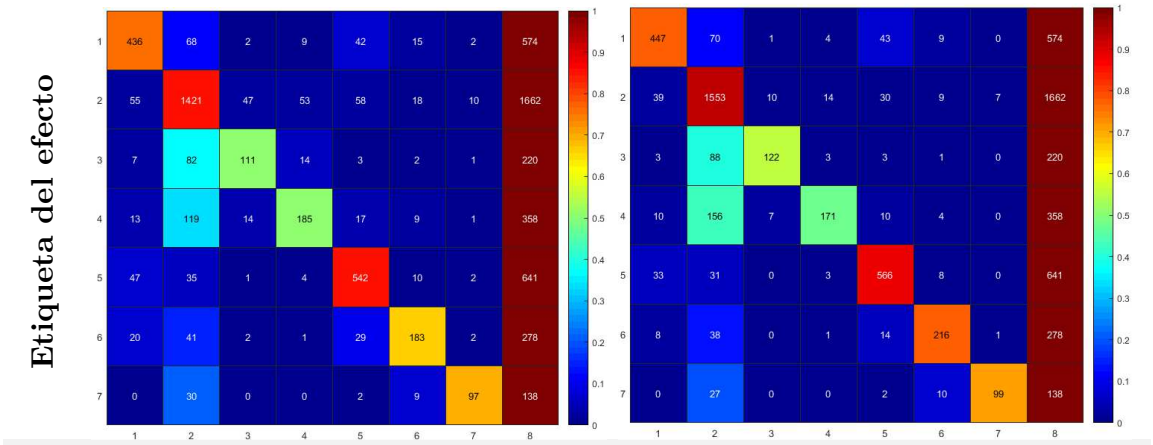


Figura 5.11 Matriz de confusión de reconocimiento de efectos, antes y después de la mejora.

Para este caso se puede notar que nuevamente la mayoría de los efectos presentan mejoras al pasar por la BN, en especial el efecto PG (cambio permanente, girando el objeto) presenta la mayor mejora con un 11.87% de los casos. Por otro lado, el efecto TZG (cambio transitorio con un desplazamiento en el eje Z más un giro del objeto) no mejora, por el contrario, empeora el reconocimiento del efecto al pasar por la BN en un 3.91% de los casos.

Tabla 5.4 Mejoramiento del reconocimiento de efectos

Efectos									
Clases de efectos		**Ejemplos por clase**	**Efecto CR al inicio**	**Empieza CR, termina CR**	**Empieza CR, termina IR**	**Empieza IR, termina CR**	**Empieza IR, termina IR**	**Efecto CR al final**	
E1	E0	574	**436**	**424**	**12**	**23**	**115**	↑	**447**
E2	TZ	1662	**1421**	**1409**	**12**	**144**	**97**	↑	**1553**
E3	TZXY	220	**111**	**97**	**14**	**25**	**84**	↑	**122**
E4	TZG	358	**185**	**153**	**32**	**18**	**155**	↓	**171**
E5	PZ	641	**542**	**537**	**5**	**29**	**70**	↑	**566**
E6	PG	278	**183**	**176**	**7**	**40**	**55**	↑	**216**
E7	PM	138	**97**	**93**	**4**	**6**	**35**	↑	**99**
Suma		**3871**	**2975**	**2889**	**86**	**285**	**611**		**3174**
Porcentaje		**100%**	**76.85%**	**74.63%**	**2.22%**	**7.36%**	**15.78%**		**81.99%**

El análisis presentado en esta sección demuestra que de forma general el uso de la BN sí mejora la capacidad de reconocimiento de los objetos, las acciones y los efectos. Pero también se presentan algunos casos particulares en los cuales el reconocimiento no mejora, sino que por el contrario empeora. Adicionalmente con este análisis que diferencia unas clases de otras se puede determinar cuáles son las clases más beneficiadas con el uso de la BN.

5.6 Inferencia con información faltante

Uno de los atributos importantes otorgados por el uso de la BN en el modelado de los ofrecimientos, es la capacidad de lidiar con la información faltante, esto se puede agrupar en varios casos, que van desde la falta total de información de las variables de entrada hasta el mejoramiento completo cuando se tiene información de las 3 variables de entrada y se mejor el reconocimiento de todas las variables al relacionar las estimaciones iniciales que se tienen.

Para demostrar la capacidad de inferencia del modelo ante información faltante, primero vamos a recordar que, en el modelo establecido, las variables de entrada a la BN son las estimaciones iniciales (A', O' y E') dadas por 3 CNNs y las variables de salida son (A, O y E) que representan las estimaciones (mejoradas) de acciones, objetos y efectos. Para el desarrollo de esta prueba primero se obtuvieron las estimaciones iniciales de (A', O' y E'), calculadas en las CNNs, que se presentan en la fila dos de la Tabla 5.5.

Después se van tomando de a una variable inicial y se obtiene los tres reconocimientos mejorados parcialmente, primero se evalúa cuando se tiene información de (A'), es decir P(A,O,E|A'); luego cuando se tiene información de (O'), es decir P(A,O,E|O'); después se evaluaron cuando se tiene información de (E'), es decir P(A,O,E|E'). Luego se van tomando de a dos variables iniciales y se obtienen nuevamente los tres reconocimientos mejorados parcialmente, primero se evalúa cuando no se tiene información de (A'), es decir P(A,O,E|O',E'); luego cuando no se tiene información de (O'), es decir P(A,O,E|A',E'); después se evaluaron cuando no se tiene información de (E'), es decir P(A,O,E|A',O'). Por último se toman las tres variables de entrada y se realiza un reconocimiento mejorado completo, es decir P(A,O,E|A',O',E'), como se presenta en la última fila de la Tabla 5.5.

Tabla 5.5 Reconocimiento bajo incertidumbre

Variables de entrada	**Variable estimada**			**Tipo de reconocimiento**
	A	**O**	**E**	
	26.17%	8.76%	42.93%	**Sin informacion**
-	71.23%	85.02%	77.04%	**Inicial (CNN)**
A	**70.84%**	20.66%	63.41%	**Mejora parcial con una variables**
O	25.70%	**85.22%**	42.93%	
E	57.74%	16.12%	**75.40%**	
O'.E'	**63.50%**	85.64%	77.90%	**Mejora parcial con dos variables**
A'.E'	76.34%	**24.66%**	79.34%	
A'.O'	74.60%	86.29%	**76.59%**	
A'.O'.E'	**79.71%**	**86.68%**	**82.05%**	**Mejorado completo**

Para empezar, vamos a analizar la primera fila, en este caso no se tiene información de ninguna de las variables de entrada por lo tanto las estimaciones de salida tendrán siempre la variable con mayor probabilidad, que serán la acción ***agarrar*** con un 26.17% de acierto, el objeto ***jarra*** con un 8.76% de acierto y el efecto ***TZ*** con un 42.93% de acierto.

Después se toma la estimación inicial de cada variable individualmente y se calculan los reconocimientos mejorados de las tres variables de salida para cada caso. De este modo se obtienen los porcentajes de acierto que se reportan en las filas 3, 4 y 5 de la Tabla 5.5.

Luego se toman de a 2 variables de entrada y se calculan todas las salidas obteniendo las filas 6, 7 y 8 este análisis se divide en dos partes. Por un lado, se toman los casos de la diagonal principal donde; primero se presenta la estimación de (A) cuando no se tiene información de (A') dado por P(A|O',E'), donde se obtiene un porcentaje de acierto de 63.50% esto nos indica que la relación objeto-efecto es bastante discriminante para reconocer la acción realizada; segundo se realiza la estimación de (O) cuando no se tiene información de (O') dado por P(O|A',E'), lo cual nos entrega un acierto promedio del 24.66%, que nos indica que la relación acción-efecto es poco discriminante para reconocer el objeto afectado; tercero se obtiene la estimación de (E) cuando no se tiene información de (E') dado por P(E|A',O') esto entrega un

porcentaje de acierto del 76.59% y nos indica que la relación objeto-efecto es muy buena discriminante para reconocer el efecto causado. Por el otro lado, se analizan los demás elementos donde se puede apreciar que, en todos los casos, el uso de la información adicional es útil en el mejoramiento del reconocimiento, pero siempre la mayor mejora se presenta cuando se tiene los tres elementos iniciales como se aprecia en la última fila de la tabla llegando a un 79.71% en el reconocimiento de las acciones, un 86.68% en el reconocimiento de los objetos y un 82.05% en el reconocimiento de los efectos.

Las tareas de inferencia cuando no se tiene información de una variable, son de gran utilidad en problemas de manipulación robótica. La selección de un objeto consiste en que se tiene información de la acción que se desea realizar y el efecto que se desea generar, de este modo el agente selecciona el objeto a manipular. La planeación de acciones consiste en que se tiene información del objeto a manipular y el efecto que se desea causar, de este modo el agente planea la acción a realizar. En la predicción de efectos se tiene información del objeto a manipular y la acción a realizar, así se puede predecir cuál será el efecto causado.

En general se puede decir que la BN tiene la capacidad de lidiar con la falta de información, ya que se mantiene un buen porcentaje de acierto en el reconocimiento aún con la ausencia de una de las variables como se aprecia en la Tabla 5.5. Por otro lado, el tener relativamente altos índices de inferencia para acciones y efectos, es muy relevante en el área de robótica e ilustra una forma computacional de cómo implementar los ofrecimientos de los objetos en las diferentes tareas de inferencia.

5.7 Resumen

Este capítulo se divide en 2 partes. Por un lado, se tienen los experimentos y por el otro lado, se tiene el análisis de los resultados obtenidos. En cuanto a los experimentos, inicialmente se establecieron los parámetros de tamaño de lote en 64 y el número de pasos de entrenamiento en 2000 para las CNNs. Después se probaron diferentes formas de calcular las CPTs de la BN y se analizó el comportamiento de la metodología para conjuntos de entrenamiento más pequeños en la mejora del reconocimiento de acciones, objetos y efectos. Por último, se analizó la capacidad de la BN para realizar tareas de inferencia con información faltante.

Capítulo 6

Conclusiones y Trabajo futuro

Manteniendo la mira puesta en entender mejor los procesos de manipulación robótica de objetos en entornos humanos, en este proyecto de tesis se diseñó una metodología novedosa presentada en la sección 4, que descompone una interacción entre un humano y un objeto, en sus tres partes constituyentes (acción, objeto y efecto), las cuales se reconocen inicialmente en 3 CNNs. Éstas, luego se fusionan en una BN que modela el concepto de los ofrecimientos para mejorar la capacidad de reconocimiento del sistema, además de permitir lidiar con problemas de información faltante y desempeñar tareas de inferencia como se observó en la sección 5.4.

6.1 Conclusiones

A partir de cada video de la base de datos CERTH-SOR3D se obtuvo como se presenta en la sección 4.2, una imagen segmentada sin fondo y dos imágenes de flujo óptico acumulado, la primera de la mano y la segunda del objeto. La información obtenida de cada video en estas 3 imágenes es lo suficientemente discriminante ya que al utilizarse como entrada a las 3 CNNs de reconocimiento permite realizar las estimaciones de acciones, objetos y efectos con probabilidades de acierto superiores al 70% para cada caso.

Se plantearon las arquitecturas de las CNNs compuestas de 5 capas, presentadas en la sección 4.3, las cuales se entrenaron, para obtener los reconocimientos iniciales de objetos, acciones y efectos, alcanzó un 85.02%, 71.23% y 77.04% de acierto promedio respectivamente. Estas estimaciones entregadas por las CNNs fueron de gran utilidad

como entradas a la BN que realizó las tareas de mejora del reconocimiento e inferencia.

Se estableció la BN compuesta de tres sub-redes independientes que se presenta en la sección 4.4 para codificar los ofrecimientos de las relaciones entre los objetos las acciones y los efectos. La BN planteada cumple con su objetivo al obtener mejoras en el reconocimiento de hasta 2.35%, 10.10% y 6.35% respectivamente para objetos, acciones y efectos.

El mejoramiento en el reconocimiento de los objetos es la única tarea en la que nos podemos comparar con otros autores y aunque solo tenemos una mejora del 1,71% contra la mejora del 4.31% reportado en (Thermos, et al., 2017). Por otro lado, la metodología planteada tiene varios puntos a favor ya que también se mejora simultáneamente el reconocimiento de las otras variables presentes en la interacción y adicionalmente el modelo planteado se puede utilizar en tareas como selección de objetos, planeación de acciones y predicción de efectos.

La BN planteada en la metodología es capaz de lidiar con los problemas de información faltante de alguna variable, presentando siempre una mejora ante la adición de otra variable. Además, es útil en tareas como la selección del objeto a manipular alcanzando un 63.50% de acierto, planeación de la acción a realizar alcanzando un 24.66% de acierto y predicción del efecto causado alcanzando un 76.59% de acierto. Los desempeños alcanzados en las diferentes tareas de inferencia muestran la utilidad de la metodología planteada en tareas relacionadas con la manipulación robótica de objetos.

6.2 Contribuciones

1. Un modelo novedoso de los ofrecimientos que relacionan acciones, objetos y efectos reconocidos en 3 redes neuronales convolucionales, combinados en una red bayesiana.
2. Mostrar experimentalmente que el modelo propuesto mejora el reconocimiento de objetos, acciones y efectos cuando se tienen estimaciones iniciales de interés y se cambian con la información de otra de las variables o cambiando las tres variables en conjunto.

3. Mostrar experimentalmente que es posible inferir objetos, acciones o efectos, aunque no se tenga información de ellos, usando información de las otras dos variables.
4. Incorporar información de efectos a una base de datos que contenía relaciones entre objetos y acciones.

6.3 Trabajo Futuro

Inicialmente se puede llevar a cabo una mejora de la metodología planteada mejorando los modelos de clasificación inicial de objetos, acciones y efectos. Esto se puede atacar por dos frentes, primero extrayendo información que pueda ser más discriminante que los flujos ópticos que tenemos actualmente en el reconocimiento de las acciones y los efectos, por otro lado, se pueden implementar CNNs y/o BNs más sofisticadas.

Por otro lado, en el desarrollo de este proyecto se ha establecido un modelo de los ofrecimientos que le permite a un robot, entender las relaciones entre los objetos, las acciones y los efectos de las interacciones entre un sujeto y un objeto, pero no se ha utilizado este conocimiento en tareas de manipulación robótica propiamente dicha, por lo tanto, se puede usar la información del modelo para poder realizar planeación de tareas de manipulación.

Otro paso a seguir sería plantear una metodología que le permita a un robot aprender a realizar tareas con objetos, de forma similar a como lo hacen los seres humanos, donde primero un maestro muestra cómo se deben realizar las tareas, definiendo los objetos, las acciones y los efectos presentes, para después pedirle al aprendiz que intente realizar por su propia mano las tareas que observó, de este modo el robot puede determinar cuánto de lo aprendido en los modelos por demostración, es útil para él al momento de la ejecución. Además, se puede establecer si existen diferencia entre las habilidades del maestro y el aprendiz, para de este modo delimitar las tareas que sí puede llevar a cabo el robot.

Una vez terminada la etapa de experimentación con el mundo real, el robot adquiere la capacidad de utilizar su mapa de conocimiento en tareas de inferencia, de este modo, puede seleccionar los objetos a manipular con mayor facilidad, puede planear acciones, que sí puede llevar a cabo y predecir los efectos generados por el mismo al interactuar con el mundo.

Bibliografía

Beauchemin, S. S. & Barron, J. L., 1995. The computation of optical flow. *ACM computing surveys (CSUR),* 27(3), pp. 433--466.

Chow, C. K. & Liu, C. N., 1968. Approximating probability distributions with dependence trees. *IEEE Transactions on Information Theory,* 14((3)), p. 462–468.

Chow, K. C. & Liu, N. C., s.f. (1968). Approximating probability distributions with dependence trees.. *IEEE Transactions on Information Theory,,* 14(3), p. 462–468.

Chu, V., Akgun, B. & Thomaz, A. L., 2016. Learning haptic affordances from demonstration and human-guided exploration. *IEEE Haptics Symposium (HAPTICS),* pp. 119-125.

Chu, V., Fitzgerald, T. & Thomaz, A. L., 2016. Learning object affordances by leveraging the combination of human-guidance and self-exploration. *ACM/IEEE International Conference on Human-Robot Interaction (HRI),* Volumen 11th, pp. 221-228.

Chu, V. & Thomas, A., 2017. Analyzing differences between teachers when learning object affordances via guided exploration. *I. J. Robotics Res,* Issue 36, pp. 739-758.

Chu, V. & Thomaz, A. L., 2014. Understanding the Role of Haptics in Affordances..

Chu, V. & Thomaz, A. L., 2016. Learning and grounding haptic affordances using demonstration and human-guided exploration. *ACM/IEEE International Conference on Human-Robot Interaction (HRI),* Volumen 11th, pp. 605-606.

Ciresan, D., Ueli, M. & Jürgen, S., 2012. Multi-column deep neural networks for image classification. *2012 IEEE Conference on Computer Vision and Pattern Recognition,* pp. 3642-3649.

Eaton, D. & Murphy, K. P., 2007. Exact Bayesian structure learning from uncertain interventions. *Artificial Intelligence and Statistics,* pp. 107--114.

Feichtenhofer, C., Pinz, A. & Zisserman, A., 2016. Convolutional Two-Stream Network Fusion for Video Action Recognition. *IEEE Conference on Computer Vision and Pattern Reco.*

Friedman, N., Geiger, D. & Goldszmidt, M., 1997. Bayesian Network Classifiers. *Machine Learning 29,* pp. 131-163.

Friedman, N., Linial, M., Nachman, I. & Pér, D., 2000. Using Bayesian networks to analyze expression data. *Journal of computational biology : a journal of computational molecular cell biology,* 7 3-4(20), pp. 601-20.

Fukushima, K., 1980. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics 36,* pp. 193-202.

Gibson, J. J., 1978. The ecological approach to the visual perception of pictures. *Leonardo.*

Heckerman, D., 2008. A Tutorial on Learning with Bayesian Networks. *Innovations in Bayesian Networks.*

Heckerman, D., Geiger, D. & Chickering, D. M., 1995. Learning Bayesian networks: The combination of knowledge and statistical data. *Machine learning, Springer,* 20(3), pp. 197--243.

Humphreys, G. W. & Bruce, V., 1989. Visual cognition: Computational, experimental and europsychological perspectives. *Psychology Press.*

Jover, J. L., 1987. Ecología perceptiva: aportaciones y limitaciones. *Anuario de psicología/The UB Journal of psychology,* Volumen 36, pp. 21--40.

Koppula, H. S., 2015. Anticipating the Future by Constructing Human Activities using Object Affordances.

Koppula, H. S., Gupta, R. & Saxena, A., 2012. Human Activity Learning using Object Affordances from RGB-D Videos. *arXiv preprint arXiv:1208.0967.*

Koppula, H. S., Gupta, R. & Saxena, A., 2013. Learning human activities and object affordances from RGB-D videos. *I. J. Robotics Res.,* Volumen 32, pp. 951-970.

Koppula, H. S. & Saxena, A., 2013. Anticipating Human Activities Using Object Affordances for Reactive Robotic Response. *IEEE Transactions on Pattern Analysis and Machine Intelligence,* Volumen 38, pp. 14-29.

Koppula, H. S. & Saxena, A., 2014. Physically Grounded Spatio-temporal Object Affordances. *ECCV.*

Krizhevsky, A., Ilya, S. & Hinton, G. E., 2012. ImageNet Classification with Deep Convolutional Neural Networks. *Commun. ACM,* Volumen 60, pp. 84-90.

LeCun, Y., BOTTOU, L. & Haffner, P., 1998. Gradient-Based Learning Applied to Document Recognition.

Linda, G. & Shapiro, C. G., 2001. Stockman, Computer vision. *Prentice Hall, Upper Saddle River, NJ.*

Lopes, M., Melo, F. S. & Montesano, L., 2007. Affordance-based imitation learning in robots. *IEEE/RSJ International Conference on Intelligent Robots and Systems,* pp. 1015-1021.

Moldovan, B., Moreno, P. & Otterlo, M. v., 2013. On the use of probabilistic relational affordance models for sequential manipulation tasks in robotics. *IEEE International Conference on Robotics and Automation,* pp. 1290-1295.

Moldovan, B. y otros, 2012. Learning relational affordance models for robots in multi-object manipulation tasks. *IEEE International Conference on Robotics and Automation.*

Moldovan, B. y otros, 2011. Statistical Relational Learning of Object Affordances for Robotic Manipulation.

Moldovan, B. & Raedt, L. D., 2014-a. Occluded object search by relational affordances. *IEEE International Conference on Robotics and Automation (ICRA),* pp. 169-174.

Moldovan, B. & Raedt, L. D., 2014-b. Learning relational affordance models for two-arm robots. *IEEE/RSJ International Conference on Intelligent Robots and Systems,* pp. 2916-2922.

Montesano, L. & Lopes, M., 2009. Learning grasping affordances from local visual descriptors. *IEEE 8th International Conference on Development and Learning,* pp. 1-6.

Montesano, L., Lopes, M., Bernardino, A. & Santos-Victor, J., 2007. Affordances, development and imitation. *IEEE 6th International Conference on Development and Learning,* pp. 270-275.

Montesano, L., Lopes, M., Bernardino, A. & Santos-Victor, J., 2007. Modeling affordances using Bayesian networks. *IEEE/RSJ International Conference on Intelligent Robots and Systems,* pp. 4102-4107.

Montesano, L., Lopes, M., Bernardino, A. & Santos-Victor, J., 2008. Learning Object Affordances: From Sensory--Motor Coordination to Imitation.. *IEEE Transactions on Robotics 24,* pp. 15-26.

Norman, D. A., 1988. The Psychology of Everyday Things, Basic Books. *Basic Books.*

Nwana, H. S., 1996. Software agents: An overview. *The knowledge engineering review, Cambridge University Press,* 11(3), pp. 205--244.

Pearl, J., 2002. Causality : Models , Reasoning , and Inference.

Rebane, G. & Pearl, J., 1987. The Recovery of Causal Poly-Trees from Statistical Data. *Int. J. Approx. Reasoning,* 2(341).

Royden, C. S. & Moore, K. D., 2012. Use of speed cues in the detection of moving objects by moving observers. *Vision research, Elsevier,* Volumen 59, pp. 17--24.

Schermer, B. W., 2007. *Software agents, surveillance, and the right to privacy: a legislative framework for agent-enabled surveillance.* s.l.:Leiden University Press.

Simonyan, K. & Zisserman, A., 2014. Two-Stream Convolutional Networks for Action Recognition in Videos. En: *NIPS.* s.l.:s.n.

Spirtes, P. & Glymour, C., 2016. An Algorithm for Fast Recovery of Sparse Causal Graphs.

Sucar, L., 2008. Clasificadores Bayesianos: De Datos a Conceptos. *European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML PKDD 2008).*

Sucar, L. E., 2015. *Probabilistic Graphical Models: Principles and Applications.* s.l.:Springer Publishing Company, Incorporated.

Thermos, S., Papadopoulos, G. T., Daras, P. & Potamianos, G., 2017. Deep Affordance-Grounded Sensorimotor Object Recognition. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR),* pp. 49-57.

Zhang, G. & Chanson, H., 2018. • Application of local optical flow methods to high-velocity free-surface flows: Validation and application to stepped chutes. *Experimental Thermal and Fluid Science,* Volumen 90, pp. 186--199.

Zunt, D., 2002. Who did actually invent the word “robot” and what does it mean. *The Karel Capek website.*

Anexos

A. Porcentaje de entrenamiento a utilizar

En este anexo se explica cómo se dividió el conjunto de entrenamiento de la base de datos, para realizar entrenamientos parciales de la red y así poder determinar la capacidad de aprendizaje de la red compuesta, ante un número menor de ejemplos de entrenamiento.

Para empezar, se debe recordar que la base de datos está dividida en función a los sujetos, esto quiere decir que un sujeto y todas las acciones que realiza se toman como un solo paquete de información. El conjunto de entrenamiento está compuesto de 98 sujetos de los cuales a cada uno se les asignó aleatoriamente un número de 1 a 5, de este modo a los que se les asignó el número 1, conforman un subconjunto llamado parte 1 que contiene aproximadamente el 20% del conjunto de total de entrenamiento, de modo similar se formaron las demás partes.

Total	Entrenamiento (100%)				
Partes	1(20%)	2(20%)	3(20%)	4(20%)	5(20%)

Tabla 8.1 División de la base de datos en 5 partes

A continuación, se presentan las tablas donde se reportan los porcentajes de acierto sobre el conjunto de validación para las redes aprendidas con un subconjunto de entrenamiento. En la primera columna se anotan las partes tomadas para cada entrenamiento, en la columna dos se presenta el porcentaje de acierto de la CNN reconocedora de las acciones, la 3 presenta el delta de mejora al pasar por la BN, en la columna 4 se tiene el porcentaje acierto de la BN con el reconocimiento mejorado de las acciones y de modo similar se reportan los objetos y los efectos. Aquí se debe tener en cuenta que cada experimento se realizó 5 veces con paquetes diferentes de ejemplos de entrenamiento que son los que se indican en la primera columna.

Tabla 8.2 Porcentajes de acierto en el reconocimiento inicial y mejorado de acciones, objetos y efectos para subconjuntos del 20%

Partes	A'	∆A	Am	O'	∆O	Om	E'	∆E	Em
1	58.41%	7.26%	65.67%	68.51%	2.20%	70.71%	62.34%	6.17%	68.51%
2	54.69%	10.13%	64.82%	65.72%	4.65%	70.37%	63.14%	4.26%	67.40%
3	57.53%	10.69%	68.23%	77.63%	1.39%	79.02%	65.38%	5.48%	70.86%
4	49.16%	12.74%	61.90%	74.27%	1.55%	75.82%	61.84%	5.71%	67.55%
5	54.97%	9.40%	64.38%	68.33%	1.60%	69.93%	61.92%	5.79%	67.71%
Media	54.75%	9.71%	64.93%	70.63%	1.88%	73.00%	62.90%	5.39%	68.38%
DesvEst	3.61%	1.99%	2.29%	4.89%	1.36%	4.05%	1.47%	0.73%	1.44%

Tabla 8.3 Porcentajes de acierto en el reconocimiento inicial y mejorado de acciones, objetos y efectos para subconjuntos del 40%

Partes	A'	∆A	Am	O'	∆O	Om	E'	∆E	Em
1,2	57.84%	7.34%	65.18%	68.36%	2.77%	71.13%	60.40%	7.03%	67.43%
2,3	57.38%	8.14%	65.51%	66.29%	4.47%	70.76%	60.27%	7.00%	67.27%
3,4	60.76%	8.91%	69.67%	76.98%	2.82%	79.80%	68.20%	4.60%	72.80%
4,5	58.25%	8.42%	66.68%	74.94%	2.07%	77.01%	62.67%	6.51%	69.18%
5,1	54.35%	10.10%	64.45%	62.70%	3.00%	65.69%	60.35%	7.18%	67.53%
Media	57.64%	8.49%	66.25%	69.45%	2.84%	72.54%	62.24%	6.29%	68.78%
DesvEst	2.29%	1.02%	2.05%	5.98%	0.88%	5.57%	3.41%	1.07%	2.34%

Tabla 8.4 Porcentajes de acierto en el reconocimiento inicial y mejorado de acciones, objetos y efectos para subconjuntos del 60%

Partes	A'	∆A	Am	O'	∆O	Om	E'	∆E	Em
1,2,3	63.60%	8.11%	71.71%	72.95%	2.74%	75.69%	69.16%	5.86%	75.02%
2,3,4	65.23%	8.73%	73.96%	81.25%	2.92%	84.16%	71.45%	5.22%	76.67%
3,4,5	66.70%	8.81%	75.51%	83.70%	1.45%	85.15%	72.49%	5.35%	77.84%
4,5,1	64.09%	8.86%	72.95%	80.78%	1.29%	82.07%	69.47%	6.64%	76.10%
5,1,2	63.96%	6.61%	70.58%	74.14%	1.34%	75.48%	67.48%	5.92%	73.39%
Media	64.70%	8.12%	72.90%	78.33%	1.71%	80.30%	69.96%	5.75%	75.77%
DesvEst	1.27%	0.95%	1.92%	4.73%	0.81%	4.63%	1.98%	0.56%	1.69%

Tabla 8.5 Porcentajes de acierto en el reconocimiento inicial y mejorado de acciones, objetos y efectos para subconjuntos del 80%

Partes	A'	∆A	Am	O'	∆O	Om	E'	∆E	Em
1,2,3,4	68.43%	6.85%	75.28%	81.48%	2.69%	84.16%	70.73%	7.36%	78.09%
2,3,4,5	69.28%	8.78%	78.07%	84.27%	1.81%	86.08%	74.58%	5.24%	79.82%
3,4,5,1	70.50%	8.96%	79.46%	85.90%	0.93%	86.83%	73.96%	6.77%	80.73%
4,5,1,2	68.20%	7.52%	75.72%	85.02%	1.89%	86.90%	73.62%	5.92%	79.54%
5,1,2,3	65.33%	6.77%	72.10%	76.03%	2.79%	78.82%	70.16%	5.01%	75.17%
Media	68.31%	7.67%	76.04%	82.38%	1.73%	84.44%	72.57%	5.93%	78.62%
DesvEst	1.91%	1.05%	2.83%	4.00%	0.76%	3.39%	2.02%	1.00%	2.17%

Tabla 8.6 Porcentajes de acierto en el reconocimiento inicial y mejorado de acciones, objetos y efectos para subconjuntos del 100%

Partes	A'	∆A	Am	O'	∆O	Om	E'	∆E	Em
1,2,3,4,5	72.69%	6.85%	79.54%	84.11%	2.12%	86.23%	76.85%	5.14%	81.99%
1,2,3,4,5	69.98%	10.10%	80.08%	84.86%	1.24%	86.10%	76.83%	5.06%	81.89%
1,2,3,4,5	71.07%	8.45%	79.51%	86.10%	1.63%	87.73%	77.45%	4.83%	82.28%
Media	71.23%	8.25%	79.71%	85.02%	1.58%	86.68%	77.04%	5.01%	82.05%
DesvEst	1.37%	1.63%	0.32%	1.00%	0.44%	0.90%	0.35%	0.16%	0.20%

B. Tablas de Probabilidad Condicional CPTs

La definición de una red bayesiana consiste de dos partes, la estructura o grafo y las tablas de probabilidad condicional asociadas a dicho grafo. En este apéndice se presentan diferentes CPTs obtenidos de diferentes formas, para demostrar que las ventajas ofrecidas por la metodología planteada no dependen de los parámetros.

Basada en el conjunto de datos

Las primeras CPTs obtenidas dependen solo de la posición de los datos, de este modo se toma una relación entre dos variables como la ocurrencia de un evento, luego se normaliza en función a la variable número 1 y se obtiene una primera CPT, luego se toman los valores iguales a 0 y se cambian por 0.1, en este punto se deben ajustar las demás probabilidades para que la suma de probabilidades de la variable2 dada la variable1 sean iguales a 1

CPTs para la red de las acciones

Primero se presenta la disposición de las acciones en función a las 54 interacciones

Tabla 8.7 a) Disposición de las acciones en la base de datos, b) P(A)

a)

	A1	A2	A3	A4	A5	A6	A7	A8	A9	A10	A11	A12	A13
	Cut	Grasp	Hamm	Lift	Open	Paint	Pour	Push	Rotate	Squee	Type	Unloc	Write
	2	14	2	14	2	1	2	9	4	1	1	1	1

b)

	A1	A2	A3	A4	A5	A6	A7	A8	A9	A10	A11	A12	A13
P(A)	Cut	Grasp	Hamm	Lift	Open	Paint	Pour	Push	Rotate	Squee	Type	Unloc	Write
	0.037	0.259	0.037	0.259	0.037	0.019	0.037	0.167	0.074	0.019	0.019	0.019	0.019

Tabla 8.8 a) Disposición de las relaciones acciones-objeto en la base de datos. b) P(O|A). c) P(O|A) suavizado.

a)

		O1	O2	O3	O4	O5	O6	O7	O8	O9	O10	O11	O12	O13	O14
		Ball	Book	Bottle	Box	Brush	Can	Cup	Hamm	Key	Knife	Pen	Pitcher	Smartp	Sponge
A1	Cut									1	1				
A2	Grasp	1	1	1	1	1	1	1	1	1	1	1	1	1	1
A3	Hammer		1						1						
A4	Lift	1	1	1	1	1	1	1	1	1	1	1	1	1	1
A5	Open		1		1										
A6	Paint					1									
A7	Pour			1									1		
A8	Push	1	1	1	1		1	1					1	1	1
A9	Rotate				1			1					1		1
A10	Squeeze														1
A11	Type													1	
A12	Unlock									1					
A13	Write											1			

b)

P(O|A)		O1	O2	O3	O4	O5	O6	O7	O8	O9	O10	O11	O12	O13	O14
		Ball	Book	Bottle	Box	Brush	Can	Cup	Hamm	Key	Knife	Pen	Pitcher	Smartp	Sponge
A1	Cut	0	0	0	0	0	0	0	0	0.5	0.5	0	0	0	0
A2	Grasp	0.071	0.071	0.071	0.071	0.071	0.071	0.071	0.071	0.071	0.071	0.071	0.071	0.071	0.071
A3	Hamm	0	0.5	0	0	0	0	0	0.5	0	0	0	0	0	0
A4	Lift	0.071	0.071	0.071	0.071	0.071	0.071	0.071	0.071	0.071	0.071	0.071	0.071	0.071	0.071
A5	Open	0	0.5	0	0.5	0	0	0	0	0	0	0	0	0	0
A6	Paint	0	0	0	0	1	0	0	0	0	0	0	0	0	0
A7	Pour	0	0	0.5	0	0	0	0	0	0	0	0	0.5	0	0
A8	Push	0.111	0.111	0.111	0.111	0	0.111	0.111	0	0	0	0	0.111	0.111	0.111
A9	Rotate	0	0	0	0.25	0	0	0.25	0	0	0	0	0.25	0	0.25
A10	Squee	0	0	0	0	0	0	0	0	0	0	0	0	0	1
A11	Type	0	0	0	0	0	0	0	0	0	0	0	0	1	0
A12	Unloc	0	0	0	0	0	0	0	0	1	0	0	0	0	0
A13	Write	0	0	0	0	0	0	0	0	0	0	1	0	0	0

c)

P(O|A)		O1	O2	O3	O4	O5	O6	O7	O8	O9	O10	O11	O12	O13	O14
		Ball	Book	Bottle	Box	Brush	Can	Cup	Hamm	Key	Knife	Pen	Pitcher	Smartp	Sponge
A1	Cut	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.44	0.44	0.01	0.01	0.01	0.01
A2	Grasp	0.071	0.071	0.071	0.071	0.071	0.071	0.071	0.071	0.071	0.071	0.071	0.071	0.071	0.071
A3	Hamm	0.01	0.44	0.01	0.01	0.01	0.01	0.01	0.44	0.01	0.01	0.01	0.01	0.01	0.01
A4	Lift	0.071	0.071	0.071	0.071	0.071	0.071	0.071	0.071	0.071	0.071	0.071	0.071	0.071	0.071
A5	Open	0.01	0.44	0.01	0.44	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
A6	Paint	0.01	0.01	0.01	0.01	0.87	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
A7	Pour	0.01	0.01	0.44	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.44	0.01	0.01
A8	Push	0.106	0.106	0.106	0.106	0.01	0.106	0.106	0.01	0.01	0.01	0.01	0.106	0.106	0.106
A9	Rotate	0.01	0.01	0.01	0.225	0.01	0.01	0.225	0.01	0.01	0.01	0.01	0.225	0.01	0.225
A10	Squee	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.87
A11	Type	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.87	0.01
A12	Unloc	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.87	0.01	0.01	0.01	0.01	0.01
A13	Write	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.87	0.01	0.01	0.01

Tabla 8.9 a) Disposición de las relaciones acción-acción en la base de datos. b) P(A|A). c) P(A|A) suavizado.

a)

		A1	A2	A3	A4	A5	A6	A7	A8	A9	A10	A11	A12	A13
		Cut	Grasp	Hamm	Lift	Open	Paint	Pour	Push	Rotate	Squee	Type	Unloc	Write
A1	Cut	2												
A2	Grasp		14											
A3	Hammer			2										
A4	Lift				14									
A5	Open					2								
A6	Paint						1							
A7	Pour							2						
A8	Push								9					
A9	Rotate									4				
A10	Squeeze										1			
A11	Type											1		
A12	Unlock												1	
A13	Write													1

b)

P(A|A)		A1	A2	A3	A4	A5	A6	A7	A8	A9	A10	A11	A12	A13
		Cut	Grasp	Hamm	Lift	Open	Paint	Pour	Push	Rotate	Squee	Type	Unloc	Write
A1	Cut	1	0	0	0	0	0	0	0	0	0	0	0	0
A2	Grasp	0	1	0	0	0	0	0	0	0	0	0	0	0
A3	Hamm	0	0	1	0	0	0	0	0	0	0	0	0	0
A4	Lift	0	0	0	1	0	0	0	0	0	0	0	0	0
A5	Open	0	0	0	0	1	0	0	0	0	0	0	0	0
A6	Paint	0	0	0	0	0	1	0	0	0	0	0	0	0
A7	Pour	0	0	0	0	0	0	1	0	0	0	0	0	0
A8	Push	0	0	0	0	0	0	0	1	0	0	0	0	0
A9	Rotate	0	0	0	0	0	0	0	0	1	0	0	0	0
A10	Squee	0	0	0	0	0	0	0	0	0	1	0	0	0
A11	Type	0	0	0	0	0	0	0	0	0	0	1	0	0
A12	Unloc	0	0	0	0	0	0	0	0	0	0	0	1	0
A13	Write	0	0	0	0	0	0	0	0	0	0	0	0	1

c)

P(A|A)		A1	A2	A3	A4	A5	A6	A7	A8	A9	A10	A11	A12	A13
		Cut	Grasp	Hamm	Lift	Open	Paint	Pour	Push	Rotate	Squee	Type	Unloc	Write
A1	Cut	0.88	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
A2	Grasp	0.01	0.88	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
A3	Hamm	0.01	0.01	0.88	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
A4	Lift	0.01	0.01	0.01	0.88	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
A5	Open	0.01	0.01	0.01	0.01	0.88	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
A6	Paint	0.01	0.01	0.01	0.01	0.01	0.88	0.01	0.01	0.01	0.01	0.01	0.01	0.01
A7	Pour	0.01	0.01	0.01	0.01	0.01	0.01	0.88	0.01	0.01	0.01	0.01	0.01	0.01
A8	Push	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.88	0.01	0.01	0.01	0.01	0.01
A9	Rotate	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.88	0.01	0.01	0.01	0.01
A10	Squee	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.88	0.01	0.01	0.01
A11	Type	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.88	0.01	0.01
A12	Unloc	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.88	0.01
A13	Write	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.88

Tabla 8.10 a) Disposición de las relaciones acción-efecto en la base de datos. b) P(E|A). c) P(E|A) suavizado.

a)

		E1	E2	E3	E4	E5	E6	E7
		E0	TZ	TZXY	TZG	PZ	PG	PM
A1	Cut			2				
A2	Grasp	7	7					
A3	Hammer		2					
A4	Lift		14					
A5	Open							2
A6	Paint			1				
A7	Pour			2				
A8	Push					9		
A9	Rotate						4	
A10	Squeeze				1			
A11	Type	1						
A12	Unlock				1			
A13	Write				1			

b)

P(E|A)		E1	E2	E3	E4	E5	E6	E7
		E0	TZ	TZXY	TZG	PZ	PG	PM
A1	Cut	0	0	1	0	0	0	0
A2	Grasp	0.5	0.5	0	0	0	0	0
A3	Hamm	0	1	0	0	0	0	0
A4	Lift	0	1	0	0	0	0	0
A5	Open	0	0	0	0	0	0	1
A6	Paint	0	0	1	0	0	0	0
A7	Pour	0	0	1	0	0	0	0
A8	Push	0	0	0	0	1	0	0
A9	Rotate	0	0	0	0	0	1	0
A10	Squee	0	0	0	1	0	0	0
A11	Type	1	0	0	0	0	0	0
A12	Unloc	0	0	0	1	0	0	0
A13	Write	0	0	0	1	0	0	0

c)

P(E|A)		E1	E2	E3	E4	E5	E6	E7
		E0	TZ	TZXY	TZG	PZ	PG	PM
A1	Cut	0.01	0.01	0.94	0.01	0.01	0.01	0.01
A2	Grasp	0.475	0.475	0.01	0.01	0.01	0.01	0.01
A3	Hamm	0.01	0.94	0.01	0.01	0.01	0.01	0.01
A4	Lift	0.01	0.94	0.01	0.01	0.01	0.01	0.01
A5	Open	0.01	0.01	0.01	0.01	0.01	0.01	0.94
A6	Paint	0.01	0.01	0.94	0.01	0.01	0.01	0.01
A7	Pour	0.01	0.01	0.94	0.01	0.01	0.01	0.01
A8	Push	0.01	0.01	0.01	0.01	0.94	0.01	0.01
A9	Rotate	0.01	0.01	0.01	0.01	0.01	0.94	0.01
A10	Squee	0.01	0.01	0.01	0.94	0.01	0.01	0.01
A11	Type	0.94	0.01	0.01	0.01	0.01	0.01	0.01
A12	Unloc	0.01	0.01	0.01	0.94	0.01	0.01	0.01
A13	Write	0.01	0.01	0.01	0.94	0.01	0.01	0.01

CPTs para la red de los objetos

Tabla 8.11 a) Disposición de los objetos en la base de datos. b) P(O).

a)

P(O)	O1	O2	O3	O4	O5	O6	O7	O8	O9	O10	O11	O12	O13	O14
	Ball	Book	Bottle	Box	Brush	Can	Cup	Hamm	Key	Knife	Pen	Pitcher	Smartp	Sponge
	3	5	4	5	3	3	4	3	4	3	3	5	4	5

b)

P(O)	O1	O2	O3	O4	O5	O6	O7	O8	O9	O10	O11	O12	O13	O14
	Ball	Book	Bottle	Box	Brush	Can	Cup	Hamm	Key	Knife	Pen	Pitcher	Smartp	Sponge
	0.056	0.093	0.074	0.093	0.056	0.056	0.074	0.056	0.074	0.056	0.056	0.093	0.074	0.093

Tabla 8.12 a) Disposición de las relaciones objeto-objeto en la basa de datos. b) P(O|O). c) P(O|O) suavizado.

a)

P(O|O)		O1 Ball	O2 Book	O3 Bottle	O4 Box	O5 Brush	O6 Can	O7 Cup	O8 Hamm	O9 Key	O10 Knife	O11 Pen	O12 Pitcher	O13 Smartp	O14 Sponge
O1	Ball	3													
O2	Book		5												
O3	Bottle			4											
O4	Box				5										
O5	Brush					3									
O6	Can						3								
O7	Cup							4							
O8	Hammer								3						
O9	Key									4					
O10	Knife										3				
O11	Pen											3			
O12	Pitcher												5		
O13	Smartphone													4	
O14	Sponge														5

b)

P(O|O)		O1 Ball	O2 Book	O3 Bottle	O4 Box	O5 Brush	O6 Can	O7 Cup	O8 Hamm	O9 Key	O10 Knife	O11 Pen	O12 Pitcher	O13 Smartp	O14 Sponge
O1	Ball	1	0	0	0	0	0	0	0	0	0	0	0	0	0
O2	Book	0	1	0	0	0	0	0	0	0	0	0	0	0	0
O3	Bottle	0	0	1	0	0	0	0	0	0	0	0	0	0	0
O4	Box	0	0	0	1	0	0	0	0	0	0	0	0	0	0
O5	Brush	0	0	0	0	1	0	0	0	0	0	0	0	0	0
O6	Can	0	0	0	0	0	1	0	0	0	0	0	0	0	0
O7	Cup	0	0	0	0	0	0	1	0	0	0	0	0	0	0
O8	Hamm	0	0	0	0	0	0	0	1	0	0	0	0	0	0
O9	Key	0	0	0	0	0	0	0	0	1	0	0	0	0	0
O10	Knife	0	0	0	0	0	0	0	0	0	1	0	0	0	0
O11	Pen	0	0	0	0	0	0	0	0	0	0	1	0	0	0
O12	Pitcher	0	0	0	0	0	0	0	0	0	0	0	1	0	0
O13	Smartp	0	0	0	0	0	0	0	0	0	0	0	0	1	0
O14	Sponge	0	0	0	0	0	0	0	0	0	0	0	0	0	1

c)

P(O|O)		O1 Ball	O2 Book	O3 Bottle	O4 Box	O5 Brush	O6 Can	O7 Cup	O8 Hamm	O9 Key	O10 Knife	O11 Pen	O12 Pitcher	O13 Smartp	O14 Sponge
O1	Ball	0.87	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
O2	Book	0.01	0.87	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
O3	Bottle	0.01	0.01	0.87	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
O4	Box	0.01	0.01	0.01	0.87	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
O5	Brush	0.01	0.01	0.01	0.01	0.87	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
O6	Can	0.01	0.01	0.01	0.01	0.01	0.87	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
O7	Cup	0.01	0.01	0.01	0.01	0.01	0.01	0.87	0.01	0.01	0.01	0.01	0.01	0.01	0.01
O8	Hamm	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.87	0.01	0.01	0.01	0.01	0.01	0.01
O9	Key	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.87	0.01	0.01	0.01	0.01	0.01
O10	Knife	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.87	0.01	0.01	0.01	0.01
O11	Pen	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.87	0.01	0.01	0.01
O12	Pitcher	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.87	0.01	0.01
O13	Smartp	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.87	0.01
O14	Sponge	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.87

Tabla 8.13 a) Disposición de las relaciones objeto-acción en la base de datos. b) P(A|O). c) P(A|O) suavizado.

a)

P(A|O)		A1 Cut	A2 Grasp	A3 Hamm	A4 Lift	A5 Open	A6 Paint	A7 Pour	A8 Push	A9 Rotate	A10 Squee	A11 Type	A12 Unloc	A13 Write
O1	Ball		1		1				1					
O2	Book		1	1	1	1			1					
O3	Bottle		1		1			1	1					
O4	Box		1		1	1			1	1				
O5	Brush		1		1		1							
O6	Can		1		1				1					
O7	Cup		1		1				1	1				
O8	Hammer		1	1	1									
O9	Key	1	1		1								1	
O10	Knife	1	1		1									
O11	Pen		1		1									1
O12	Pitcher		1		1			1	1	1				
O13	Smartphone		1		1				1			1		
O14	Sponge		1		1				1	1	1			

b)

P(A|O)		A1 Cut	A2 Grasp	A3 Hamm	A4 Lift	A5 Open	A6 Paint	A7 Pour	A8 Push	A9 Rotate	A10 Squee	A11 Type	A12 Unloc	A13 Write
O1	Ball	0	0.333	0	0.333	0	0	0	0.333	0	0	0	0	0
O2	Book	0	0.2	0.2	0.2	0.2	0	0	0.2	0	0	0	0	0
O3	Bottle	0	0.25	0	0.25	0	0	0.25	0.25	0	0	0	0	0
O4	Box	0	0.2	0	0.2	0.2	0	0	0.2	0.2	0	0	0	0
O5	Brush	0	0.333	0	0.333	0	0.333	0	0	0	0	0	0	0
O6	Can	0	0.333	0	0.333	0	0	0	0.333	0	0	0	0	0
O7	Cup	0	0.25	0	0.25	0	0	0	0.25	0.25	0	0	0	0
O8	Hamm	0	0.333	0.333	0.333	0	0	0	0	0	0	0	0	0
O9	Key	0.25	0.25	0	0.25	0	0	0	0	0	0	0	0.25	0
O10	Knife	0.333	0.333	0	0.333	0	0	0	0	0	0	0	0	0
O11	Pen	0	0.333	0	0.333	0	0	0	0	0	0	0	0	0.333
O12	Pitcher	0	0.2	0	0.2	0	0	0.2	0.2	0.2	0	0	0	0
O13	Smartp	0	0.25	0	0.25	0	0	0	0.25	0	0	0.25	0	0
O14	Sponge	0	0.2	0	0.2	0	0	0	0.2	0.2	0.2	0	0	0

c)

O8		A1 Cut	A2 Grasp	A3 Hamm	A4 Lift	A5 Open	A6 Paint	A7 Pour	A8 Push	A9 Rotate	A10 Squee	A11 Type	A12 Unloc	A13 Write
O1	Ball	0.01	0.3	0.01	0.3	0.01	0.01	0.01	0.3	0.01	0.01	0.01	0.01	0.01
O2	Book	0.01	0.184	0.184	0.184	0.184	0.01	0.01	0.184	0.01	0.01	0.01	0.01	0.01
O3	Bottle	0.01	0.228	0.01	0.228	0.01	0.01	0.228	0.228	0.01	0.01	0.01	0.01	0.01
O4	Box	0.01	0.184	0.01	0.184	0.184	0.01	0.01	0.184	0.184	0.01	0.01	0.01	0.01
O5	Brush	0.01	0.3	0.01	0.3	0.01	0.3	0.01	0.01	0.01	0.01	0.01	0.01	0.01
O6	Can	0.01	0.3	0.01	0.3	0.01	0.01	0.01	0.3	0.01	0.01	0.01	0.01	0.01
O7	Cup	0.01	0.228	0.01	0.228	0.01	0.01	0.01	0.228	0.228	0.01	0.01	0.01	0.01
O8	Hamm	0.01	0.3	0.3	0.3	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
O9	Key	0.228	0.228	0.01	0.228	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.228	0.01
O10	Knife	0.3	0.3	0.01	0.3	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
O11	Pen	0.01	0.3	0.01	0.3	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.3
O12	Pitcher	0.01	0.184	0.01	0.184	0.01	0.01	0.184	0.184	0.184	0.01	0.01	0.01	0.01
O13	Smartp	0.01	0.228	0.01	0.228	0.01	0.01	0.01	0.228	0.01	0.01	0.228	0.01	0.01
O14	Sponge	0.01	0.184	0.01	0.184	0.01	0.01	0.01	0.184	0.184	0.184	0.01	0.01	0.01

Tabla 8.14 a) Disposición de las relaciones objeto-objeto en la base de datos. b) P(E|O). c) P(E|O) suavizado.

a)

P(E|O)		E1 E0	E2 TZ	E3 TZXY	E4 TZG	E5 PZ	E6 PG	E7 PM
O1	Ball	1	1			1		
O2	Book		3			1		1
O3	Bottle	1	1		1	1		
O4	Box	1	1		1	1		1
O5	Brush		2	1				
O6	Can	1	1			1		
O7	Cup	1	1			1	1	
O8	Hammer		3					
O9	Key		1	2	1			
O10	Knife		2	1				
O11	Pen		2		1			
O12	Pitcher	1	1		1	1	1	
O13	Smartp	1	2			1		
O14	Sponge	1	1		1	1	1	

b)

P(E|O)		E1 E0	E2 TZ	E3 TZXY	E4 TZG	E5 PZ	E6 PG	E7 PM
O1	Ball	0.333	0.333	0	0	0.333	0	0
O2	Book	0	0.6	0	0	0.2	0	0.2
O3	Bottle	0.25	0.25	0	0.25	0.25	0	0
O4	Box	0.2	0.2	0	0.2	0.2	0	0.2
O5	Brush	0	0.667	0.333	0	0	0	0
O6	Can	0.333	0.333	0	0	0.333	0	0
O7	Cup	0.25	0.25	0	0	0.25	0.25	0
O8	Hamm	0	1	0	0	0	0	0
O9	Key	0	0.25	0.5	0.25	0	0	0
O10	Knife	0	0.667	0.333	0	0	0	0
O11	Pen	0	0.667	0	0.333	0	0	0
O12	Pitcher	0.2	0.2	0	0.2	0.2	0.2	0
O13	Smartp	0.25	0.5	0	0	0.25	0	0
O14	Sponge	0.2	0.2	0	0.2	0.2	0.2	0

c)

P(E|O)		E1 E0	E2 TZ	E3 TZXY	E4 TZG	E5 PZ	E6 PG	E7 PM
O1	Ball	0.32	0.32	0.01	0.01	0.32	0.01	0.01
O2	Book	0.01	0.568	0.01	0.01	0.196	0.01	0.196
O3	Bottle	0.243	0.243	0.01	0.243	0.243	0.01	0.01
O4	Box	0.196	0.196	0.01	0.196	0.196	0.01	0.196
O5	Brush	0.01	0.63	0.32	0.01	0.01	0.01	0.01
O6	Can	0.32	0.32	0.01	0.01	0.32	0.01	0.01
O7	Cup	0.243	0.243	0.01	0.01	0.243	0.243	0.01
O8	Hamm	0.01	0.94	0.01	0.01	0.01	0.01	0.01
O9	Key	0.01	0.243	0.475	0.243	0.01	0.01	0.01
O10	Knife	0.01	0.63	0.32	0.01	0.01	0.01	0.01
O11	Pen	0.01	0.63	0.01	0.32	0.01	0.01	0.01
O12	Pitcher	0.196	0.196	0.01	0.196	0.196	0.196	0.01
O13	Smartp	0.243	0.475	0.01	0.01	0.243	0.01	0.01
O14	Sponge	0.196	0.196	0.01	0.196	0.196	0.196	0.01

CPTs para la red de los efectos

Tabla 8.15 a) Disposición de los efectos en la base de datos. b) P(E).

a)

P(E)	E1 E0	E2 TZ	E3 TZXY	E4 TZG	E5 PZ	E6 PG	E7 PM
	8	23	3	5	9	4	2

b)

P(E)	E1 E0	E2 TZ	E3 TZXY	E4 TZG	E5 PZ	E6 PG	E7 PM
	0.148	0.426	0.056	0.093	0.167	0.074	0.037

Tabla 8.16 a) Disposición de las relaciones efecto-objeto en la base de datos. b) P(O|E). c) P(O|E) suavizado.

a)

P(O|E)		O1 Ball	O2 Book	O3 Bottle	O4 Box	O5 Brush	O6 Can	O7 Cup	O8 Hamm	O9 Key	O10 Knife	O11 Pen	O12 Pitcher	O13 Smartp	O14 Sponge
E1	E0	1		1	1		1	1					1	1	1
E2	TZ	1	3	1	1	2	1	1	3	2	2	2	1	2	1
E3	TZXY					1				1	1				
E4	TZG			1	1					1		1			1
E5	PZ	1	1	1	1		1	1					1	1	1
E6	PG				1			1					1		1
E7	PM		1		1										

b)

P(O|E)		O1 Ball	O2 Book	O3 Bottle	O4 Box	O5 Brush	O6 Can	O7 Cup	O8 Hamm	O9 Key	O10 Knife	O11 Pen	O12 Pitcher	O13 Smartp	O14 Sponge
E1	E0	0.125	0	0.125	0.125	0	0.125	0.125	0	0	0	0	0.125	0.125	0.125
E2	TZ	0.043	0.13	0.043	0.043	0.087	0.043	0.043	0.13	0.087	0.087	0.087	0.043	0.087	0.043
E3	TZXY	0	0	0	0	0.333	0	0	0	0.333	0.333	0	0	0	0
E4	TZG	0	0	0.2	0.2	0	0	0	0	0.2	0	0.2	0	0	0.2
E5	PZ	0.111	0.111	0.111	0.111	0	0.111	0.111	0	0	0	0	0.111	0.111	0.111
E6	PG	0	0	0	0.25	0	0	0.25	0	0	0	0	0.25	0	0.25
E7	PM	0	0.5	0	0.5	0	0	0	0	0	0	0	0	0	0

c)

P(O|E)		O1 Ball	O2 Book	O3 Bottle	O4 Box	O5 Brush	O6 Can	O7 Cup	O8 Hamm	O9 Key	O10 Knife	O11 Pen	O12 Pitcher	O13 Smartp	O14 Sponge
E1	E0	0.118	0.01	0.118	0.118	0.01	0.118	0.118	0.01	0.01	0.01	0.01	0.118	0.118	0.118
E2	TZ	0.047	0.122	0.047	0.047	0.085	0.047	0.047	0.122	0.085	0.085	0.085	0.047	0.085	0.047
E3	TZXY	0.01	0.01	0.01	0.01	0.297	0.01	0.01	0.01	0.297	0.297	0.01	0.01	0.01	0.01
E4	TZG	0.01	0.01	0.182	0.182	0.01	0.01	0.01	0.01	0.182	0.01	0.182	0.01	0.01	0.182
E5	PZ	0.106	0.106	0.106	0.106	0.01	0.106	0.106	0.01	0.01	0.01	0.01	0.106	0.106	0.106
E6	PG	0.01	0.01	0.01	0.225	0.01	0.01	0.225	0.01	0.01	0.01	0.01	0.225	0.01	0.225
E7	PM	0.01	0.44	0.01	0.44	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01

Tabla 8.17 a) Disposición de las relaciones efecto-acción en la base de datos. b) P(A|E). c) P(A|E).

a)

P(A|E)		A1 Cut	A2 Grasp	A3 Hamm	A4 Lift	A5 Open	A6 Paint	A7 Pour	A8 Push	A9 Rotate	A10 Squee	A11 Type	A12 Unloc	A13 Write
E1	E0		7									1		
E2	TZ		7	2	14									
E3	TZXY	2					1							
E4	TZG							2			1		1	1
E5	PZ								9					
E6	PG									4				
E7	PM					2								

b)

P(A|E)		A1 Cut	A2 Grasp	A3 Hamm	A4 Lift	A5 Open	A6 Paint	A7 Pour	A8 Push	A9 Rotate	A10 Squee	A11 Type	A12 Unloc	A13 Write
E1	E0	0	0.875	0	0	0	0	0	0	0	0	0.125	0	0
E2	TZ	0	0.304	0.087	0.609	0	0	0	0	0	0	0	0	0
E3	TZXY	0.667	0	0	0	0	0.333	0	0	0	0	0	0	0
E4	TZG	0	0	0	0	0	0	0.4	0	0	0.2	0	0.2	0.2
E5	PZ	0	0	0	0	0	0	0	1	0	0	0	0	0
E6	PG	0	0	0	0	0	0	0	0	1	0	0	0	0
E7	PM	0	0	0	0	1	0	0	0	0	0	0	0	0

c)

P(A|E)		A1 Cut	A2 Grasp	A3 Hamm	A4 Lift	A5 Open	A6 Paint	A7 Pour	A8 Push	A9 Rotate	A10 Squee	A11 Type	A12 Unloc	A13 Write
E1	E0	0.01	0.771	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.119	0.01	0.01
E2	TZ	0.01	0.275	0.086	0.54	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
E3	TZXY	0.59	0.01	0.01	0.01	0.01	0.3	0.01	0.01	0.01	0.01	0.01	0.01	0.01
E4	TZG	0.01	0.01	0.01	0.01	0.01	0.01	0.358	0.01	0.01	0.184	0.01	0.184	0.184
E5	PZ	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.88	0.01	0.01	0.01	0.01	0.01
E6	PG	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.88	0.01	0.01	0.01	0.01
E7	PM	0.01	0.01	0.01	0.01	0.88	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01

Tabla 8.18 a) Disposición de las relaciones efecto-efecto en la base de datos. b) P(E|E). c) P(E|E) suavizado.

a)

P(E|E)		E1 E0	E2 TZ	E3 TZXY	E4 TZG	E5 PZ	E6 PG	E7 PM
E1	E0	8						
E2	TZ		23					
E3	TZXY			3				
E4	TZG				5			
E5	PZ					9		
E6	PG						4	
E7	PM							2

b)

P(E|E)		E1 E0	E2 TZ	E3 TZXY	E4 TZG	E5 PZ	E6 PG	E7 PM
E1	E0	1	0	0	0	0	0	0
E2	TZ	0	1	0	0	0	0	0
E3	TZXY	0	0	1	0	0	0	0
E4	TZG	0	0	0	1	0	0	0
E5	PZ	0	0	0	0	1	0	0
E6	PG	0	0	0	0	0	1	0
E7	PM	0	0	0	0	0	0	1

c)

P(E|E)		E1 E0	E2 TZ	E3 TZXY	E4 TZG	E5 PZ	E6 PG	E7 PM
E1	E0	0.94	0.01	0.01	0.01	0.01	0.01	0.01
E2	TZ	0.01	0.94	0.01	0.01	0.01	0.01	0.01
E3	TZXY	0.01	0.01	0.94	0.01	0.01	0.01	0.01
E4	TZG	0.01	0.01	0.01	0.94	0.01	0.01	0.01
E5	PZ	0.01	0.01	0.01	0.01	0.94	0.01	0.01
E6	PG	0.01	0.01	0.01	0.01	0.01	0.94	0.01
E7	PM	0.01	0.01	0.01	0.01	0.01	0.01	0.94

Basada en estimaciones del conjunto de entrenamiento

Después se calculan otros 2 conjunto de CPTs pero esta vez se basan en los datos de entrenamiento. Primero se calcula con las estimaciones llamadas "suaves", donde se van acumulando los vectores de estimación en las respectivas tablas. Luego se hace lo mismo, pero con las estimaciones "duras" es decir que se le asigna 1 a la probabilidad mayor y las demás quedan en 0.

CPTs para la red de las acciones

Tabla 8.19 a) P(A) basada en estimaciones suaves. b) P(A) basada en estimaciones duras.

a)

P(A)	A1 Cut	A2 Grasp	A3 Hamme	A4 Lift	A5 Open	A6 Paint	A7 Pour	A8 Push	A9 Rotate	A10 Squeeze	A11 Type	A12 Unlock	A13 Write
	0.0422	0.2356	0.0422	0.2356	0.0422	0.0261	0.0422	0.155	0.0744	0.0261	0.0261	0.0261	0.0261

b)

P(A)	A1 Cut	A2 Grasp	A3 Hamm	A4 Lift	A5 Open	A6 Paint	A7 Pour	A8 Push	A9 Rotate	A10 Squee	A11 Type	A12 Unloc	A13 Write
	0.042	0.236	0.042	0.236	0.042	0.026	0.042	0.155	0.074	0.026	0.026	0.026	0.026

Tabla 8.20 a) P(O|A) basada en estimaciones suaves. b) P(O|A) basada en estimaciones duras.

a)

P(O|A)		O1 Ball	O2 Book	O3 Bottle	O4 Box	O5 Brush	O6 Can	O7 Cup	O8 Hammer	O9 Key	O10 Knife	O11 Pen	O12 Pitcher	O13 Smartph	O14 Sponge
A1	Cut	0.0347	0.0104	0.0115	0.0103	0.0351	0.0108	0.0112	0.0119	0.2756	0.3301	0.2184	0.0102	0.029	0.0107
A2	Grasp	0.056	0.0693	0.0747	0.0724	0.0636	0.0617	0.0762	0.0755	0.0922	0.0672	0.0733	0.0726	0.0891	0.0664
A3	Hamme	0.0124	0.3857	0.0174	0.0165	0.05	0.0105	0.0183	0.3672	0.0104	0.0159	0.0103	0.0101	0.0669	0.0182
A4	Lift	0.0555	0.0678	0.0708	0.0737	0.0635	0.064	0.0804	0.0756	0.091	0.0683	0.0719	0.0703	0.0873	0.0699
A5	Open	0.0118	0.401	0.0109	0.3926	0.0196	0.0172	0.0245	0.023	0.0114	0.0115	0.0111	0.0102	0.0335	0.0319
A6	Paint	0.0113	0.0179	0.0112	0.0123	0.538	0.0109	0.0108	0.0921	0.016	0.0834	0.0136	0.01	0.1721	0.0103
A7	Pour	0.01	0.0101	0.4124	0.0104	0.0103	0.0262	0.0235	0.0175	0.01	0.0107	0.01	0.4303	0.0186	0.0101
A8	Push	0.0688	0.0967	0.1138	0.1062	0.0176	0.0951	0.1175	0.0278	0.0257	0.0162	0.0229	0.1041	0.101	0.0966
A9	Rotate	0.0244	0.0208	0.0222	0.2188	0.0118	0.0133	0.2292	0.0189	0.013	0.0105	0.0122	0.2141	0.011	0.1897
A10	Squeeze	0.0571	0.0152	0.0106	0.0826	0.0118	0.0113	0.0453	0.0146	0.0144	0.011	0.0136	0.0108	0.011	0.7007
A11	Type	0.011	0.0133	0.0255	0.0114	0.0485	0.0147	0.0103	0.0439	0.0169	0.0299	0.0153	0.0102	0.749	0.0102
A12	Unlock	0.0489	0.0102	0.0104	0.0101	0.0122	0.0103	0.0106	0.0103	0.4309	0.0778	0.3455	0.01	0.0123	0.0106
A13	Write	0.0446	0.0103	0.0141	0.0108	0.0122	0.0164	0.0238	0.0106	0.4476	0.0497	0.3346	0.0102	0.0135	0.0118

b)

P(O|A)		O1 Ball	O2 Book	O3 Bottle	O4 Box	O5 Brush	O6 Can	O7 Cup	O8 Hamm	O9 Key	O10 Knife	O11 Pen	O12 Pitcher	O13 Smartp	O14 Sponge
A1	Cut	0.01	0.01	0.01	0.01	0.024	0.01	0.01	0.01	0.47	0.362	0.053	0.01	0.01	0.01
A2	Grasp	0.053	0.068	0.07	0.072	0.063	0.066	0.078	0.075	0.154	0.065	0.018	0.072	0.086	0.069
A3	Hamm	0.01	0.391	0.017	0.01	0.039	0.01	0.017	0.405	0.01	0.01	0.01	0.01	0.06	0.01
A4	Lift	0.057	0.066	0.067	0.072	0.066	0.068	0.08	0.077	0.153	0.066	0.013	0.07	0.083	0.074
A5	Open	0.01	0.402	0.01	0.438	0.024	0.01	0.017	0.01	0.01	0.01	0.01	0.01	0.024	0.024
A6	Paint	0.01	0.01	0.01	0.01	0.609	0.01	0.01	0.081	0.01	0.081	0.01	0.01	0.138	0.01
A7	Pour	0.01	0.01	0.424	0.01	0.01	0.024	0.024	0.017	0.01	0.01	0.01	0.424	0.017	0.01
A8	Push	0.075	0.097	0.109	0.106	0.016	0.102	0.123	0.021	0.029	0.013	0.01	0.101	0.102	0.104
A9	Rotate	0.01	0.021	0.018	0.224	0.01	0.01	0.239	0.014	0.014	0.01	0.01	0.216	0.01	0.205
A10	Squee	0.025	0.01	0.01	0.085	0.01	0.01	0.025	0.01	0.01	0.01	0.01	0.01	0.01	0.775
A11	Type	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.025	0.025	0.01	0.01	0.851	0.01
A12	Unloc	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.764	0.01	0.126	0.01	0.01	0.01
A13	Write	0.01	0.01	0.01	0.01	0.01	0.025	0.025	0.01	0.837	0.01	0.025	0.01	0.01	0.01

Tabla 8.21 a) P(A|A) basada en estimaciones suaves. b) P(A|A) basada en estimaciones duras.

a)

P(A|A)		A1 Cut	A2 Grasp	A3 Hamme	A4 Lift	A5 Open	A6 Paint	A7 Pour	A8 Push	A9 Rotate	A10 Squeeze	A11 Type	A12 Unlock	A13 Write
A1	Cut	0.5197	0.1496	0.0201	0.0324	0.012	0.0114	0.0102	0.0591	0.0358	0.0103	0.0252	0.0312	0.083
A2	Grasp	0.0436	0.6347	0.0237	0.0583	0.022	0.0123	0.0114	0.0968	0.0235	0.0169	0.0197	0.0184	0.0186
A3	Hamme	0.0171	0.0904	0.6223	0.0627	0.0484	0.0274	0.0151	0.0467	0.0126	0.012	0.0136	0.0108	0.0209
A4	Lift	0.0144	0.0693	0.0236	0.715	0.0219	0.0177	0.0216	0.0244	0.0311	0.016	0.0108	0.0228	0.0114
A5	Open	0.0108	0.1116	0.0233	0.0839	0.5258	0.0165	0.0136	0.0678	0.0987	0.0105	0.0132	0.011	0.0133
A6	Paint	0.0114	0.0564	0.0159	0.1337	0.0196	0.6702	0.0145	0.0114	0.0196	0.0103	0.01	0.0165	0.0104
A7	Pour	0.0172	0.0879	0.012	0.234	0.0134	0.0104	0.4809	0.0137	0.043	0.0138	0.0103	0.053	0.0102
A8	Push	0.0293	0.0906	0.0116	0.02	0.0309	0.01	0.0108	0.6958	0.0314	0.0108	0.0361	0.0108	0.0119
A9	Rotate	0.025	0.0726	0.0164	0.1617	0.0315	0.0117	0.022	0.0427	0.5615	0.0127	0.0118	0.0143	0.016
A10	Squeeze	0.0238	0.1607	0.012	0.1616	0.0239	0.0103	0.0309	0.037	0.0591	0.3994	0.0157	0.0534	0.0122
A11	Type	0.0589	0.242	0.0101	0.026	0.0128	0.01	0.01	0.2012	0.0262	0.01	0.3507	0.0103	0.0319
A12	Unlock	0.0538	0.1664	0.0142	0.2206	0.0223	0.0136	0.0476	0.0345	0.0424	0.0344	0.0143	0.3054	0.0303
A13	Write	0.2378	0.2098	0.0228	0.0232	0.0158	0.0103	0.0108	0.0375	0.0306	0.013	0.0358	0.0384	0.3142

b)

P(A|A)		A1 Cut	A2 Grasp	A3 Hamm	A4 Lift	A5 Open	A6 Paint	A7 Pour	A8 Push	A9 Rotate	A10 Squee	A11 Type	A12 Unloc	A13 Write
A1	Cut	0.549	0.147	0.017	0.032	0.01	0.01	0.01	0.046	0.032	0.01	0.024	0.039	0.075
A2	Grasp	0.045	0.653	0.02	0.053	0.021	0.012	0.011	0.095	0.02	0.017	0.018	0.017	0.016
A3	Hamm	0.017	0.082	0.664	0.053	0.039	0.017	0.017	0.053	0.01	0.01	0.01	0.01	0.017
A4	Lift	0.013	0.068	0.023	0.731	0.022	0.016	0.018	0.025	0.03	0.013	0.011	0.018	0.011
A5	Open	0.01	0.11	0.01	0.081	0.559	0.017	0.017	0.06	0.096	0.01	0.01	0.01	0.01
A6	Paint	0.01	0.053	0.01	0.138	0.024	0.695	0.01	0.01	0.01	0.01	0.01	0.01	0.01
A7	Pour	0.017	0.088	0.01	0.238	0.01	0.01	0.488	0.017	0.046	0.01	0.01	0.046	0.01
A8	Push	0.028	0.085	0.012	0.016	0.028	0.01	0.01	0.715	0.031	0.01	0.034	0.012	0.01
A9	Rotate	0.025	0.075	0.018	0.151	0.033	0.014	0.021	0.041	0.571	0.014	0.01	0.014	0.014
A10	Squee	0.025	0.145	0.01	0.16	0.025	0.01	0.025	0.04	0.055	0.415	0.025	0.055	0.01
A11	Type	0.025	0.271	0.01	0.025	0.01	0.01	0.01	0.199	0.025	0.01	0.373	0.01	0.025
A12	Unloc	0.054	0.155	0.01	0.242	0.01	0.01	0.039	0.025	0.068	0.025	0.01	0.315	0.039
A13	Write	0.242	0.228	0.025	0.01	0.01	0.01	0.01	0.025	0.025	0.01	0.039	0.039	0.329

Tabla 8.22 a) P(E|A) basada en estimaciones suaves. b) P(E|A) basada en estimaciones duras.

a)

P(E|A)		E1 E0	E2 TZ	E3 TZXY	E4 TZG	E5 PZ	E6 PG	E7 PM
A1	Cut	0.1085	0.299	0.3669	0.0772	0.0557	0.021	0.0118
A2	Grasp	0.3327	0.3757	0.0564	0.0416	0.0862	0.0328	0.0146
A3	Hamme	0.0654	0.6792	0.0308	0.0312	0.0617	0.0293	0.0424
A4	Lift	0.0557	0.6534	0.0355	0.0525	0.0701	0.0356	0.0371
A5	Open	0.0532	0.1795	0.0263	0.0308	0.0867	0.0541	0.5093
A6	Paint	0.0591	0.2447	0.4098	0.0566	0.1088	0.0468	0.0141
A7	Pour	0.0937	0.1713	0.0218	0.4091	0.1621	0.0562	0.0257
A8	Push	0.0859	0.0677	0.0187	0.0256	0.6676	0.0574	0.017
A9	Rotate	0.1167	0.15	0.0247	0.0248	0.1141	0.4714	0.0382
A10	Squeeze	0.1008	0.2882	0.0392	0.3445	0.0716	0.0805	0.0152
A11	Type	0.4503	0.2023	0.0763	0.0331	0.1291	0.0331	0.0158
A12	Unlock	0.1044	0.4656	0.1425	0.152	0.0265	0.0238	0.0252
A13	Write	0.1152	0.3238	0.1539	0.2758	0.0266	0.0315	0.0132

P(E|A)		E1 E0	E2 TZ	E3 TZXY	E4 TZG	E5 PZ	E6 PG	E7 PM
A1	Cut	0.103	0.29	0.405	0.068	0.053	0.01	0.01
A2	Grasp	0.337	0.398	0.048	0.035	0.084	0.027	0.012
A3	Hamm	0.068	0.715	0.024	0.024	0.053	0.024	0.032
A4	Lift	0.045	0.697	0.029	0.036	0.072	0.03	0.033
A5	Open	0.046	0.174	0.024	0.031	0.088	0.046	0.531
A6	Paint	0.067	0.267	0.409	0.039	0.11	0.039	0.01
A7	Pour	0.067	0.16	0.017	0.452	0.167	0.053	0.024
A8	Push	0.076	0.065	0.018	0.026	0.686	0.054	0.015
A9	Rotate	0.102	0.144	0.018	0.021	0.113	0.521	0.021
A10	Squee	0.085	0.28	0.04	0.385	0.07	0.07	0.01
A11	Type	0.503	0.199	0.068	0.025	0.112	0.025	0.01
A12	Unloc	0.097	0.503	0.112	0.155	0.025	0.025	0.025
A13	Write	0.112	0.344	0.155	0.271	0.025	0.025	0.01

b)

CPTs para la red de los objetos

Tabla 8.23 a) P(O) basada en estimaciones suaves. b) P(O) basada en estimaciones duras.

P(O)	O1 Ball	O2 Book	O3 Bottle	O4 Box	O5 Brush	O6 Can	O7 Cup	O8 Hammer	O9 Key	O10 Knife	O11 Pen	O12 Pitcher	O13 Smartph	O14 Sponge
	0.0578	0.0896	0.0737	0.0896	0.0578	0.0578	0.0737	0.0578	0.0737	0.0578	0.0578	0.0896	0.0737	0.0896

a)

P(O)	O1 Ball	O2 Book	O3 Bottle	O4 Box	O5 Brush	O6 Can	O7 Cup	O8 Hammer	O9 Key	O10 Knife	O11 Pen	O12 Pitcher	O13 Smartph	O14 Sponge
	0.0578	0.0896	0.0737	0.0896	0.0578	0.0578	0.0737	0.0578	0.0737	0.0578	0.0578	0.0896	0.0737	0.0896

b)

Tabla 8.24 a) P(O|O) basada en estimaciones suaves. b) P(O|O) basada en estimaciones duras.

P(O|O)		O1 Ball	O2 Book	O3 Bottle	O4 Box	O5 Brush	O6 Can	O7 Cup	O8 Hammer	O9 Key	O10 Knife	O11 Pen	O12 Pitcher	O13 Smartph	O14 Sponge
O1	Ball	0.4857	0.0465	0.0146	0.0133	0.0196	0.0121	0.0341	0.0132	0.149	0.0272	0.1209	0.0178	0.0119	0.0339
O2	Book	0.0151	0.7526	0.0106	0.014	0.0268	0.0104	0.0313	0.0167	0.0102	0.0105	0.0101	0.0101	0.055	0.0265
O3	Bottle	0.01	0.01	0.7947	0.0105	0.0103	0.0409	0.0104	0.0228	0.01	0.0105	0.01	0.0282	0.0215	0.01
O4	Box	0.0118	0.0142	0.0114	0.7785	0.0115	0.0187	0.021	0.0261	0.0129	0.0113	0.0122	0.0102	0.0141	0.046
O5	Brush	0.0113	0.0186	0.0115	0.0157	0.5514	0.0116	0.012	0.0738	0.0135	0.0833	0.0125	0.0101	0.1638	0.0109
O6	Can	0.0104	0.0102	0.0611	0.0395	0.012	0.6978	0.0115	0.0591	0.0114	0.0194	0.0131	0.0199	0.0245	0.0101
O7	Cup	0.0236	0.0236	0.0239	0.0246	0.011	0.0127	0.7837	0.016	0.0113	0.0106	0.0111	0.0127	0.0105	0.0247
O8	Hammer	0.0132	0.0219	0.0235	0.0194	0.0721	0.0115	0.0143	0.6916	0.011	0.0191	0.0107	0.015	0.0634	0.0134
O9	Key	0.0489	0.0106	0.0138	0.0103	0.0131	0.0114	0.0119	0.0108	0.4209	0.0755	0.3344	0.0134	0.0134	0.0115
O10	Knife	0.0214	0.0108	0.0116	0.0132	0.0528	0.011	0.0137	0.0154	0.1276	0.5578	0.0974	0.0101	0.046	0.011
O11	Pen	0.0471	0.0105	0.0121	0.0105	0.0127	0.0127	0.0156	0.0107	0.4462	0.0533	0.3334	0.0102	0.0137	0.0114
O12	Pitcher	0.01	0.0101	0.0261	0.0107	0.01	0.0102	0.0379	0.01	0.01	0.01	0.01	0.8247	0.01	0.0102
O13	Smartph	0.0105	0.0124	0.017	0.0113	0.0469	0.0129	0.0103	0.0623	0.0138	0.0317	0.0134	0.0101	0.7373	0.0101
O14	Sponge	0.0538	0.014	0.0107	0.075	0.0123	0.0115	0.0443	0.0193	0.0125	0.0109	0.012	0.0136	0.0108	0.6992

a)

b)

P(O|O)		O1 Ball	O2 Book	O3 Bottle	O4 Box	O5 Brush	O6 Can	O7 Cup	O8 Hammer	O9 Key	O10 Knife	O11 Pen	O12 Pitcher	O13 Smartph	O14 Sponge
O1	Ball	0.6159	0.0442	0.0149	0.0149	0.0247	0.01	0.0149	0.01	0.1908	0.01	0.01	0.0149	0.01	0.0149
O2	Book	0.0158	0.7722	0.01	0.01	0.033	0.01	0.033	0.01	0.01	0.01	0.01	0.01	0.0445	0.0215
O3	Bottle	0.01	0.01	0.8198	0.01	0.01	0.0387	0.01	0.0243	0.01	0.01	0.01	0.01	0.0172	0.01
O4	Box	0.01	0.01	0.01	0.846	0.01	0.01	0.016	0.01	0.013	0.01	0.01	0.01	0.01	0.025
O5	Brush	0.01	0.01	0.01	0.0147	0.635	0.01	0.01	0.0664	0.01	0.057	0.01	0.01	0.1369	0.01
O6	Can	0.01	0.01	0.0148	0.0387	0.01	0.7792	0.01	0.0434	0.01	0.0243	0.01	0.0196	0.01	0.01
O7	Cup	0.01	0.0204	0.0135	0.01	0.01	0.01	0.8457	0.0135	0.0135	0.01	0.01	0.01	0.01	0.0135
O8	Hammer	0.01	0.01	0.0242	0.01	0.0525	0.01	0.01	0.766	0.01	0.01	0.01	0.0147	0.0525	0.01
O9	Key	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.7744	0.0206	0.0879	0.0171	0.01	0.01
O10	Knife	0.01	0.01	0.01	0.0147	0.024	0.01	0.0147	0.01	0.1689	0.6877	0.01	0.01	0.01	0.01
O11	Pen	0.01	0.01	0.01	0.01	0.01	0.0148	0.0148	0.01	0.8415	0.01	0.029	0.01	0.01	0.01
O12	Pitcher	0.01	0.01	0.0188	0.01	0.01	0.01	0.0395	0.01	0.01	0.01	0.01	0.8317	0.01	0.01
O13	Smartph	0.01	0.01	0.01	0.01	0.0205	0.01	0.01	0.0485	0.0135	0.0205	0.01	0.01	0.8071	0.01
O14	Sponge	0.0159	0.01	0.01	0.0452	0.01	0.01	0.0276	0.0159	0.01	0.01	0.01	0.0129	0.01	0.8025

Tabla 8.25 a) P(A|O) basada en estimaciones suaves. b) P(A|O) basada en estimaciones duras.

a)

P(A|O)		A1 Cut	A2 Grasp	A3 Hamme	A4 Lift	A5 Open	A6 Paint	A7 Pour	A8 Push	A9 Rotate	A10 Squeeze	A11 Type	A12 Unlock	A13 Write
O1	Ball	0.0149	0.2448	0.0152	0.2514	0.0654	0.0102	0.0215	0.2622	0.0223	0.0182	0.0376	0.0157	0.0105
O2	Book	0.0296	0.1736	0.1553	0.1562	0.1788	0.0132	0.0129	0.1962	0.0228	0.013	0.0138	0.0123	0.0124
O3	Bottle	0.0172	0.2337	0.0122	0.2787	0.0148	0.0111	0.0941	0.228	0.0333	0.0194	0.0114	0.025	0.0111
O4	Box	0.0169	0.2605	0.0172	0.1861	0.1053	0.0125	0.0189	0.1462	0.1745	0.0124	0.0149	0.0116	0.0129
O5	Brush	0.0106	0.2904	0.0273	0.303	0.0139	0.2522	0.0144	0.0158	0.0184	0.0105	0.0104	0.0126	0.0104
O6	Can	0.0158	0.2771	0.0104	0.2836	0.0146	0.01	0.0147	0.2923	0.0284	0.0101	0.0121	0.0106	0.0104
O7	Cup	0.0176	0.226	0.0101	0.2425	0.0143	0.01	0.0154	0.1996	0.1977	0.0102	0.022	0.0118	0.0126
O8	Hammer	0.0133	0.2751	0.2606	0.2788	0.017	0.0362	0.0136	0.0301	0.0133	0.0122	0.0101	0.0107	0.0192
O9	Key	0.219	0.1864	0.0159	0.2371	0.0149	0.0113	0.0242	0.0567	0.0339	0.0192	0.024	0.1062	0.0414
O10	Knife	0.1717	0.2798	0.0172	0.2827	0.0139	0.011	0.0151	0.0558	0.0227	0.0169	0.0206	0.0419	0.0407
O11	Pen	0.1212	0.2707	0.0177	0.2611	0.0205	0.0104	0.0169	0.0512	0.0221	0.0123	0.0256	0.0406	0.1197
O12	Pitcher	0.0112	0.1647	0.0101	0.2614	0.0172	0.01	0.1418	0.1689	0.1417	0.0178	0.0126	0.0225	0.01
O13	Smartph	0.0511	0.2403	0.0117	0.1951	0.0147	0.0104	0.0104	0.2801	0.0259	0.0136	0.1051	0.0111	0.0205
O14	Sponge	0.0188	0.2156	0.0181	0.2261	0.0313	0.0103	0.0173	0.1979	0.1006	0.0905	0.0252	0.0221	0.0161

b)

P(A|O)		A1 Cut	A2 Grasp	A3 Hamme	A4 Lift	A5 Open	A6 Paint	A7 Pour	A8 Push	A9 Rotate	A10 Squeeze	A11 Type	A12 Unlock	A13 Write
O1	Ball	0.01	0.2446	0.0149	0.2543	0.0686	0.01	0.0149	0.2739	0.0198	0.0149	0.0344	0.0198	0.01
O2	Book	0.0273	0.174	0.1596	0.1624	0.1855	0.0129	0.01	0.2027	0.0158	0.01	0.01	0.01	0.01
O3	Bottle	0.0172	0.2429	0.01	0.2788	0.0136	0.01	0.096	0.2322	0.0315	0.0172	0.01	0.0208	0.01
O4	Box	0.016	0.2587	0.013	0.1838	0.1089	0.013	0.019	0.1478	0.1838	0.01	0.016	0.01	0.01
O5	Brush	0.01	0.292	0.0194	0.3155	0.0147	0.2591	0.0147	0.0147	0.01	0.01	0.01	0.01	0.01
O6	Can	0.0148	0.2823	0.01	0.2823	0.01	0.01	0.0148	0.3014	0.0243	0.01	0.01	0.01	0.01
O7	Cup	0.0169	0.2215	0.01	0.2423	0.0135	0.01	0.0135	0.2111	0.2042	0.01	0.0169	0.01	0.01
O8	Hammer	0.0147	0.2746	0.2652	0.2888	0.0147	0.0289	0.0147	0.0289	0.01	0.01	0.01	0.01	0.0195
O9	Key	0.2365	0.1763	0.0171	0.24	0.01	0.01	0.0206	0.0454	0.0419	0.0171	0.0277	0.1091	0.0383
O10	Knife	0.1829	0.2811	0.0193	0.2904	0.01	0.01	0.0147	0.0521	0.0147	0.0193	0.0147	0.0427	0.038
O11	Pen	0.1193	0.2903	0.0195	0.2666	0.0195	0.01	0.0148	0.0528	0.0148	0.01	0.0243	0.029	0.1193
O12	Pitcher	0.01	0.1661	0.01	0.2574	0.0159	0.01	0.1425	0.172	0.1425	0.0188	0.0129	0.0218	0.01
O13	Smartph	0.0415	0.2617	0.01	0.1883	0.0135	0.01	0.01	0.2757	0.024	0.0135	0.1149	0.01	0.017
O14	Sponge	0.0188	0.2213	0.0188	0.2184	0.0335	0.01	0.0159	0.1979	0.101	0.0922	0.0247	0.0217	0.0159

Tabla 8.26 a) P(E|O) basada en estimaciones suaves. b) P(E|O) basada en estimaciones duras.

a)

P(E|O)		E1 E0	E2 TZ	E3 TZXY	E4 TZG	E5 PZ	E6 PG	E7 PM
O1	Ball	0.2841	0.2634	0.0183	0.0333	0.2421	0.0566	0.0322
O2	Book	0.0522	0.4694	0.0223	0.0237	0.1971	0.0239	0.1413
O3	Bottle	0.2	0.3061	0.0169	0.1088	0.2455	0.037	0.0157
O4	Box	0.1413	0.2424	0.0194	0.0267	0.2224	0.1521	0.1258
O5	Brush	0.0659	0.5301	0.1839	0.0386	0.0705	0.028	0.013
O6	Can	0.2426	0.2911	0.0193	0.0271	0.296	0.0377	0.0161
O7	Cup	0.2034	0.2723	0.0279	0.0201	0.2451	0.1432	0.018
O8	Hammer	0.146	0.5726	0.0442	0.0333	0.0718	0.0302	0.0319
O9	Key	0.0944	0.4555	0.1687	0.117	0.0438	0.0245	0.0261
O10	Knife	0.0948	0.4636	0.2271	0.0403	0.0614	0.0206	0.0224
O11	Pen	0.0825	0.4841	0.0985	0.1729	0.0484	0.0217	0.022
O12	Pitcher	0.1466	0.2298	0.0183	0.1385	0.2171	0.1568	0.0229
O13	Smartph	0.1862	0.3934	0.039	0.0426	0.2225	0.0265	0.02
O14	Sponge	0.1759	0.2682	0.0259	0.0908	0.215	0.1282	0.0262

b)

P(E|O)		E1 E0	E2 TZ	E3 TZXY	E4 TZG	E5 PZ	E6 PG	E7 PM
O1	Ball	0.2836	0.2788	0.0149	0.0295	0.2494	0.0491	0.0247
O2	Book	0.0474	0.4846	0.0186	0.0186	0.1941	0.0244	0.1423
O3	Bottle	0.1856	0.3182	0.0136	0.1068	0.2537	0.0387	0.0136
O4	Box	0.1329	0.2377	0.019	0.028	0.2288	0.1598	0.1239
O5	Brush	0.057	0.5645	0.1745	0.0241	0.0711	0.0288	0.01
O6	Can	0.2393	0.2967	0.0148	0.0243	0.3062	0.0339	0.0148
O7	Cup	0.2007	0.284	0.0239	0.0204	0.2493	0.1383	0.0135
O8	Hammer	0.147	0.5959	0.0336	0.0289	0.0762	0.0242	0.0242
O9	Key	0.0879	0.4913	0.1657	0.102	0.0419	0.0171	0.0242
O10	Knife	0.0848	0.4867	0.2297	0.0287	0.0614	0.0193	0.0193
O11	Pen	0.0718	0.5422	0.0908	0.143	0.048	0.0148	0.0195
O12	Pitcher	0.1514	0.225	0.0159	0.1425	0.2162	0.1573	0.0218
O13	Smartph	0.1848	0.412	0.031	0.0415	0.2233	0.0205	0.017
O14	Sponge	0.1685	0.2918	0.0217	0.0922	0.2125	0.1274	0.0159

CPTs para la red de los efectos

Tabla 8.27 a) P(E) basada en estimaciones suaves. b) P(E) basada en estimaciones duras.

a)

P(E)	E1 E0	E2 TZ	E3 TZXY	E4 TZG	E5 PZ	E6 PG	E7 PM
	0.148	0.406	0.062	0.096	0.165	0.079	0.044

b)

P(E)	E1 E0	E2 TZ	E3 TZXY	E4 TZG	E5 PZ	E6 PG	E7 PM
	0.148	0.406	0.062	0.096	0.165	0.079	0.044

Tabla 8.28 a) P(O|E) basada en estimaciones suaves. b) P(O|E) basada en estimaciones duras.

a)

P(O|E)		O1 Ball	O2 Book	O3 Bottle	O4 Box	O5 Brush	O6 Can	O7 Cup	O8 Hamme	O9 Key	O10 Knife	O11 Pen	O12 Pitcher	O13 Smartph	O14 Sponge
E1	E0	0.079	0.029	0.111	0.108	0.023	0.09	0.116	0.038	0.059	0.04	0.047	0.103	0.115	0.112
E2	TZ	0.045	0.113	0.058	0.063	0.079	0.049	0.063	0.113	0.107	0.079	0.084	0.059	0.104	0.056
E3	TZXY	0.036	0.031	0.024	0.032	0.193	0.023	0.036	0.043	0.169	0.208	0.136	0.035	0.087	0.017
E4	TZG	0.038	0.033	0.167	0.051	0.018	0.016	0.04	0.019	0.157	0.032	0.126	0.17	0.05	0.153
E5	PZ	0.068	0.101	0.12	0.116	0.023	0.092	0.119	0.033	0.03	0.023	0.026	0.115	0.106	0.099
E6	PG	0.029	0.033	0.036	0.197	0.018	0.036	0.205	0.03	0.026	0.018	0.023	0.209	0.032	0.179
E7	PM	0.019	0.346	0.027	0.334	0.023	0.029	0.053	0.042	0.037	0.031	0.03	0.034	0.034	0.031

b)

P(O|E)		O1 Ball	O2 Book	O3 Bottle	O4 Box	O5 Brush	O6 Can	O7 Cup	O8 Hamm	O9 Key	O10 Knife	O11 Pen	O12 Pitcher	O13 Smartp	O14 Sponge
E1	E0	0.081	0.026	0.104	0.11	0.02	0.098	0.116	0.032	0.093	0.038	0.01	0.102	0.122	0.118
E2	TZ	0.041	0.111	0.054	0.062	0.078	0.051	0.064	0.117	0.182	0.075	0.017	0.058	0.099	0.058
E3	TZXY	0.026	0.026	0.02	0.036	0.207	0.02	0.041	0.036	0.28	0.218	0.041	0.036	0.067	0.015
E4	TZG	0.019	0.035	0.17	0.045	0.016	0.016	0.041	0.016	0.274	0.013	0.038	0.167	0.048	0.17
E5	PZ	0.073	0.102	0.115	0.12	0.022	0.1	0.12	0.027	0.037	0.018	0.01	0.112	0.108	0.105
E6	PG	0.018	0.034	0.03	0.199	0.018	0.034	0.211	0.026	0.034	0.022	0.01	0.211	0.03	0.191
E7	PM	0.018	0.343	0.026	0.359	0.026	0.026	0.057	0.033	0.057	0.033	0.01	0.033	0.026	0.026

Tabla 8.29 a) P(A|E) basada en estimaciones suaves. b) P(A|E) basada en estimaciones duras.

a)

P(A|E)		A1 Cut	A2 Grasp	A3 Hamme	A4 Lift	A5 Open	A6 Paint	A7 Pour	A8 Push	A9 Rotate	A10 Squeeze	A11 Type	A12 Unlock	A13 Write
E1	E0	0.031	0.593	0.017	0.075	0.029	0.01	0.011	0.136	0.041	0.019	0.055	0.022	0.02
E2	TZ	0.042	0.263	0.078	0.445	0.028	0.019	0.019	0.057	0.034	0.019	0.014	0.023	0.019
E3	TZXY	0.278	0.178	0.022	0.116	0.026	0.198	0.017	0.067	0.037	0.014	0.021	0.025	0.062
E4	TZG	0.072	0.166	0.015	0.191	0.026	0.016	0.179	0.087	0.058	0.076	0.026	0.085	0.063
E5	PZ	0.034	0.106	0.015	0.077	0.037	0.021	0.032	0.618	0.045	0.011	0.039	0.012	0.013
E6	PG	0.026	0.07	0.021	0.151	0.038	0.014	0.021	0.162	0.484	0.02	0.019	0.016	0.016
E7	PM	0.013	0.12	0.035	0.197	0.44	0.021	0.03	0.048	0.093	0.014	0.014	0.022	0.011

b)

P(A|E)		A1 Cut	A2 Grasp	A3 Hamm	A4 Lift	A5 Open	A6 Paint	A7 Pour	A8 Push	A9 Rotate	A10 Squee	A11 Type	A12 Unloc	A13 Write
E1	E0	0.024	0.606	0.016	0.069	0.028	0.01	0.012	0.134	0.041	0.018	0.057	0.024	0.02
E2	TZ	0.044	0.268	0.08	0.451	0.027	0.017	0.017	0.057	0.031	0.018	0.014	0.02	0.017
E3	TZXY	0.291	0.192	0.02	0.114	0.026	0.197	0.015	0.057	0.031	0.015	0.02	0.026	0.057
E4	TZG	0.076	0.161	0.013	0.192	0.023	0.016	0.18	0.085	0.063	0.073	0.026	0.085	0.067
E5	PZ	0.032	0.1	0.015	0.074	0.035	0.022	0.032	0.636	0.044	0.01	0.037	0.012	0.01
E6	PG	0.026	0.066	0.022	0.147	0.042	0.018	0.018	0.163	0.493	0.018	0.018	0.014	0.014
E7	PM	0.01	0.134	0.026	0.196	0.467	0.018	0.033	0.041	0.088	0.01	0.01	0.018	0.01

Tabla 8.30 a) P(E|E) basada en estimaciones suaves. b) P(E|E) basada en estimaciones duras.

a)

P(E|E)		E1	E2	E3	E4	E5	E6	E7
		E0	TZ	TZXY	TZG	PZ	PG	PM
E1	E0	0.701	0.084	0.019	0.022	0.119	0.041	0.014
E2	TZ	0.069	0.727	0.052	0.051	0.051	0.029	0.02
E3	TZXY	0.055	0.292	0.515	0.082	0.025	0.019	0.012
E4	TZG	0.078	0.323	0.075	0.402	0.051	0.057	0.014
E5	PZ	0.076	0.038	0.013	0.013	0.813	0.032	0.014
E6	PG	0.107	0.119	0.014	0.021	0.107	0.595	0.038
E7	PM	0.025	0.127	0.012	0.015	0.034	0.055	0.731

b)

P(E|E)		E1	E2	E3	E4	E5	E6	E7
		E0	TZ	TZXY	TZG	PZ	PG	PM
E1	E0	0.73	0.077	0.016	0.014	0.118	0.032	0.012
E2	TZ	0.058	0.774	0.042	0.037	0.05	0.023	0.016
E3	TZXY	0.041	0.301	0.54	0.072	0.02	0.015	0.01
E4	TZG	0.063	0.327	0.067	0.431	0.048	0.054	0.01
E5	PZ	0.069	0.035	0.012	0.012	0.831	0.027	0.013
E6	PG	0.086	0.111	0.01	0.018	0.111	0.646	0.018
E7	PM	0.018	0.126	0.01	0.018	0.033	0.049	0.746

C. Evaluación de la metodología

Para la evaluación de la metodología presentada se entrenaron 3 veces las redes neuronales convolucionales del reconocimiento inicial, aunque los parámetros de entrenamiento eran iguales se obtuvieron redes ligeramente diferentes debido a las componentes de aleatoriedad. A continuación, se presentan los porcentajes de acierto para el conjunto de validación tanto en el reconocimiento inicial X' como el reconocimiento mejorado Xm, para las 4 diferentes clases de CPTs presentados en el apéndice anterior. Adicionalmente se calculan las desviaciones estándar y los promedios del entrenamiento los cuales se reportan en el cuerpo de la tesis

Tabla 8.31 Reconocimiento mejorado con las CPTs uniformes.

DataSet	A'	ΔA	Am	O'	ΔO	Om	E'	ΔE	Em
a	72.69%	6.10%	78.79%	84.11%	1.96%	86.08%	76.85%	5.68%	82.54%
b	69.98%	9.64%	79.62%	84.86%	1.45%	86.31%	76.83%	6.35%	83.18%
c	71.07%	7.93%	79.00%	86.10%	1.03%	87.14%	77.45%	5.14%	82.59%
Media	71.23%	7.62%	79.13%	85.02%	1.38%	86.50%	77.04%	5.68%	82.77%
DesvEst	1.37%	1.77%	0.43%	1.00%	0.47%	0.56%	0.35%	0.61%	0.36%

Tabla 8.32 Reconocimiento mejorado con las CPTs uniformes suavizadas.

DataSet 0.1	A'	ΔA	Am	O'	ΔO	Om	E'	ΔE	Em
a	72.69%	6.28%	78.97%	84.11%	1.86%	85.97%	76.85%	5.63%	82.49%
b	69.98%	9.43%	79.41%	84.86%	1.63%	86.49%	76.83%	5.94%	82.77%
c	71.07%	7.47%	78.53%	86.10%	1.19%	87.29%	77.45%	5.04%	82.49%
Media	71.23%	7.51%	78.97%	85.02%	1.50%	86.58%	77.04%	5.51%	82.58%
DesvEst	1.37%	1.59%	0.44%	1.00%	0.34%	0.66%	0.35%	0.46%	0.16%

Tabla 8.33 Reconocimiento mejorado con las CPTs de estimaciones suaves.

soft	A'	ΔA	Am	O'	ΔO	Om	E'	ΔE	Em
a	72.69%	6.64%	79.33%	84.11%	2.35%	86.46%	76.85%	4.99%	81.84%
b	69.98%	9.79%	79.77%	84.86%	1.47%	86.33%	76.83%	4.78%	81.61%
c	71.07%	8.21%	79.28%	86.10%	1.55%	87.65%	77.45%	4.60%	82.05%
Media	71.23%	8.01%	79.46%	85.02%	1.71%	86.81%	77.04%	4.78%	81.83%
DesvEst	1.37%	1.58%	0.27%	1.00%	0.49%	0.73%	0.35%	0.19%	0.22%

Tabla 8.34 Reconocimiento mejorado con las CPTs de estimaciones duras

hard	A'	ΔA	Am	O'	ΔO	Om	E'	ΔE	Em
a	72.69%	6.85%	79.54%	84.11%	2.12%	86.23%	76.85%	5.14%	81.99%
b	69.98%	10.10%	80.08%	84.86%	1.24%	86.10%	76.83%	5.06%	81.89%
c	71.07%	8.45%	79.51%	86.10%	1.63%	87.73%	77.45%	4.83%	82.28%
Media	71.23%	8.25%	79.71%	85.02%	1.58%	86.68%	77.04%	5.01%	82.05%
DesvEst	1.37%	1.63%	0.32%	1.00%	0.44%	0.90%	0.35%	0.16%	0.20%

D. Matrices de confusión para el análisis de las mejoras obtenidas

Para entender con mayor detalle lo que está pasando se calcula la matriz de confusión de los reconocimientos iniciales dados por las CNNs y de los reconocimientos mejorados en la BN, luego se divide en cuatro partes, primero se toman los elementos que están CR y al pasar por la red bayesiana siguen CR. Después se toman los elementos que estaban CR y al pasar por la red bayesiana, pasan a quedar IR. Luego se toman los elementos que están IR y al pasar por la red quedan CR. Por último, se miran los elementos que estaban IR y al pasar por la red siguen estando IR.

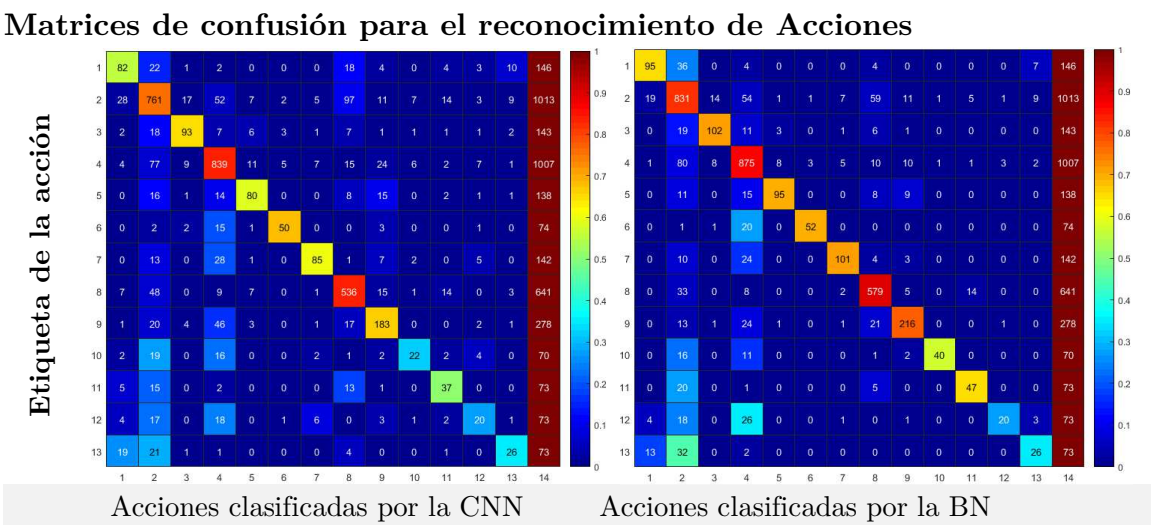


Figura 8.1 Matriz se confusión de reconocimiento de acciones, antes y después de la mejora.

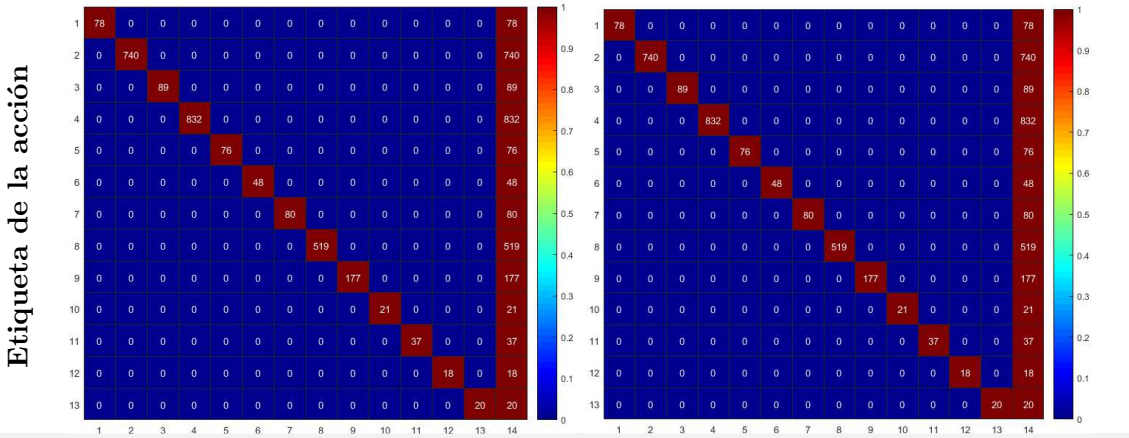


Figura 8.2 Matriz de confusión de acciones CR al inicio y CR al final.

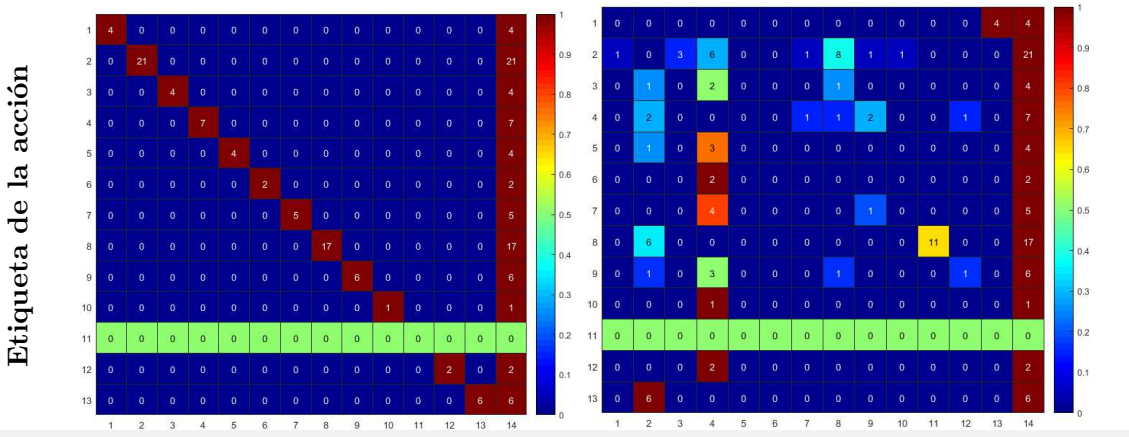


Figura 8.3 Matriz de confusión de acciones CR al inicio e IR al final.

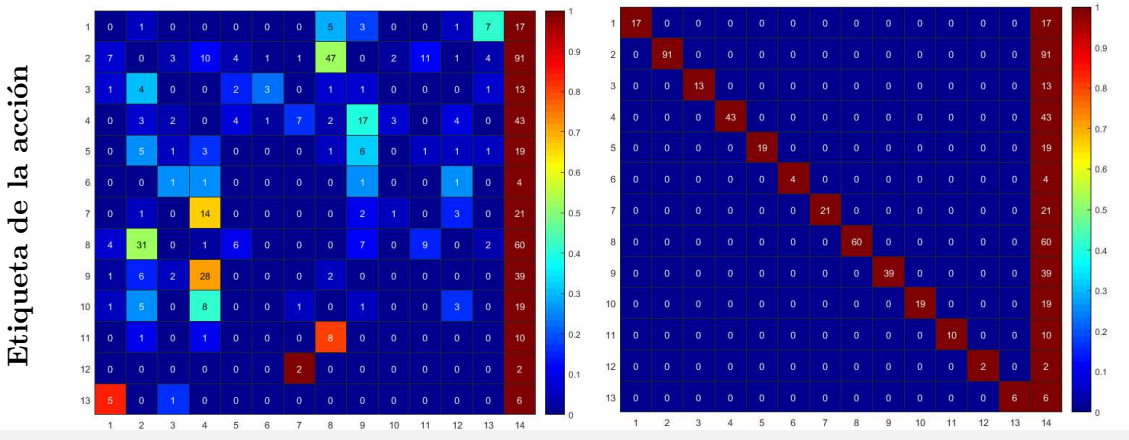


Figura 8.4 Matriz de confusión de acciones IR al inicio y CR al final.

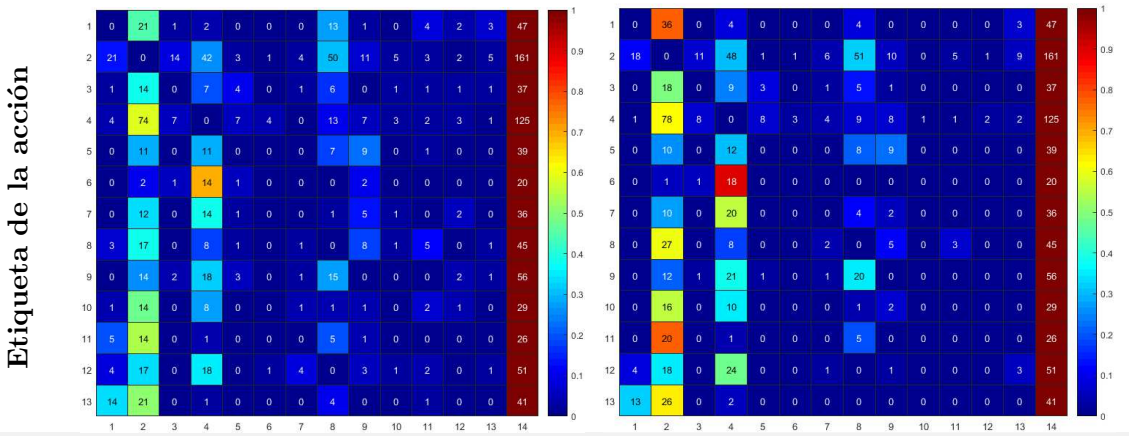


Figura 8.5 Matriz de confusión de acciones IR al inicio e IR al final.

Matrices de confusión para el reconocimiento de Objetos

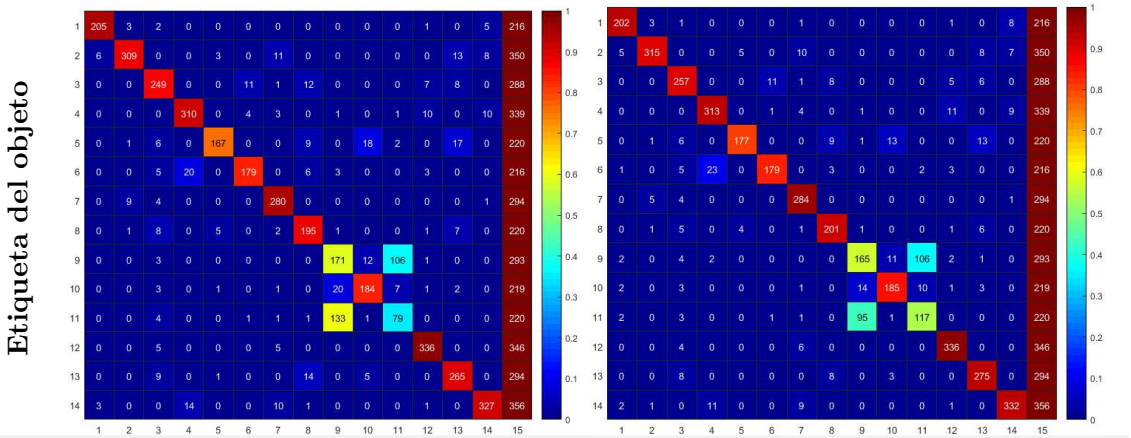


Figura 8.6 Matriz se confusión del reconocimiento de objetos, antes y después de la mejora.

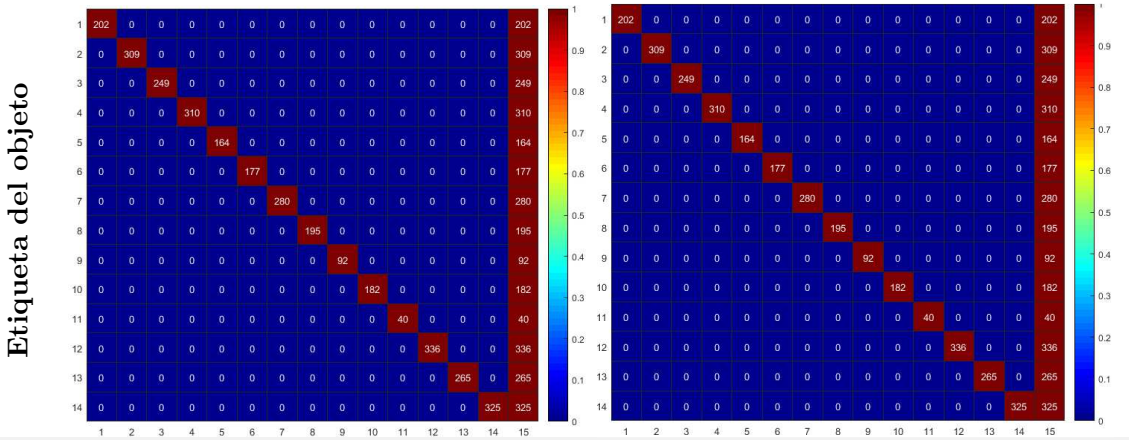


Figura 8.7 Matriz de confusión de objetos CR al inicio, CR al final.

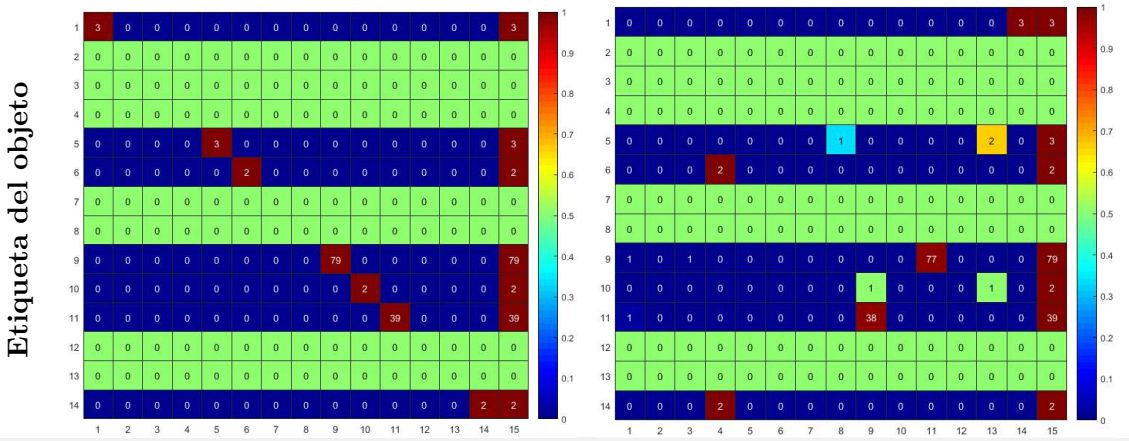


Figura 8.8 Matriz de confusión de objetos CR al inicio e IR al final.

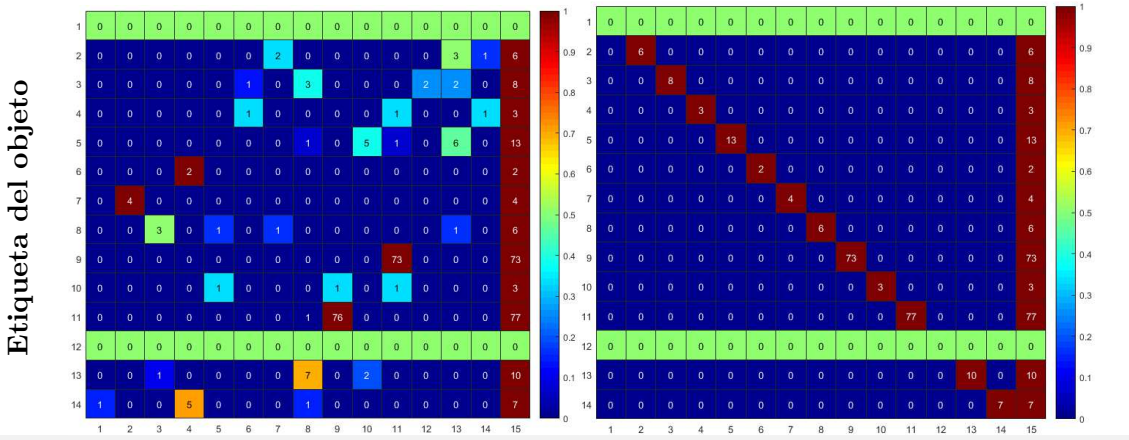


Figura 8.9 Matriz de confusión de objetos IR al inicio y CR al final.

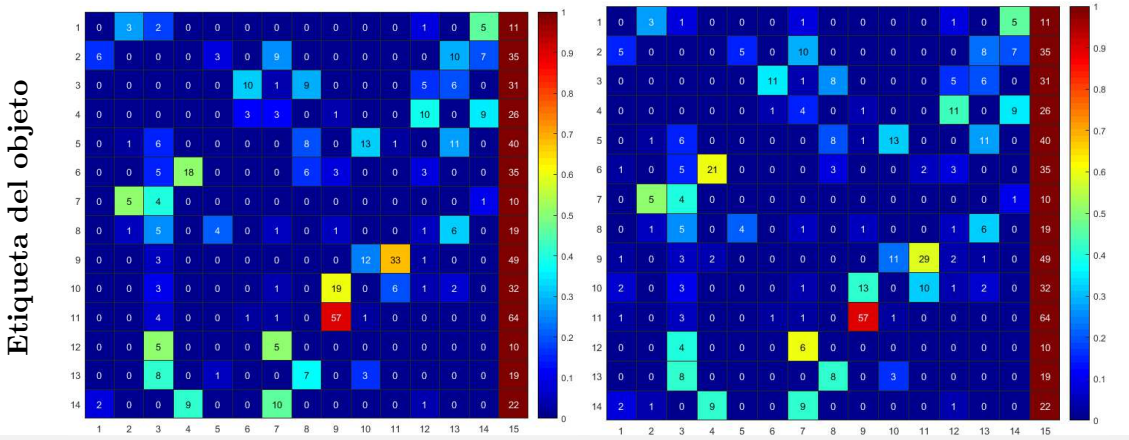


Figura 8.10 Matriz de confusión de objetos IR al inicio e IR al final.

Matrices de confusión para el reconocimiento de Efectos

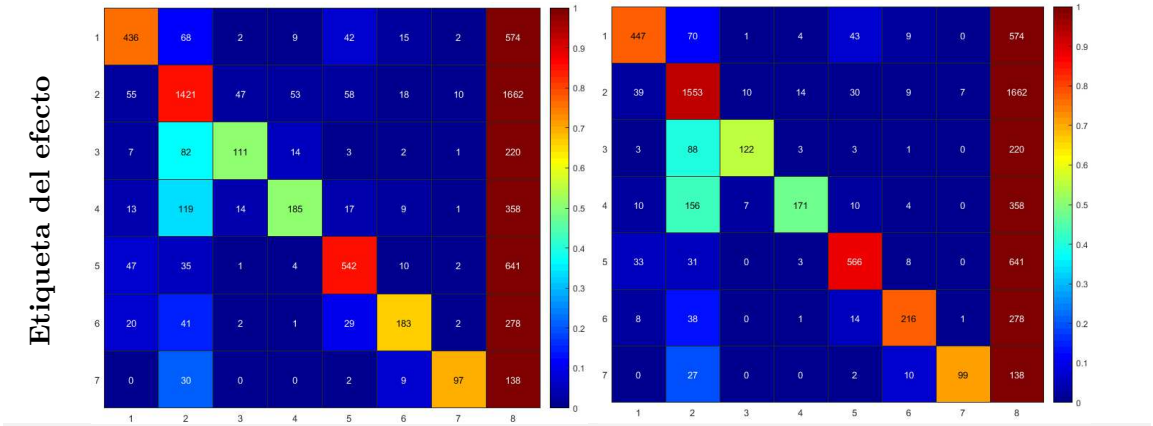


Figura 8.11 Matriz de confusión de reconocimiento de efectos, antes y después de la mejora.

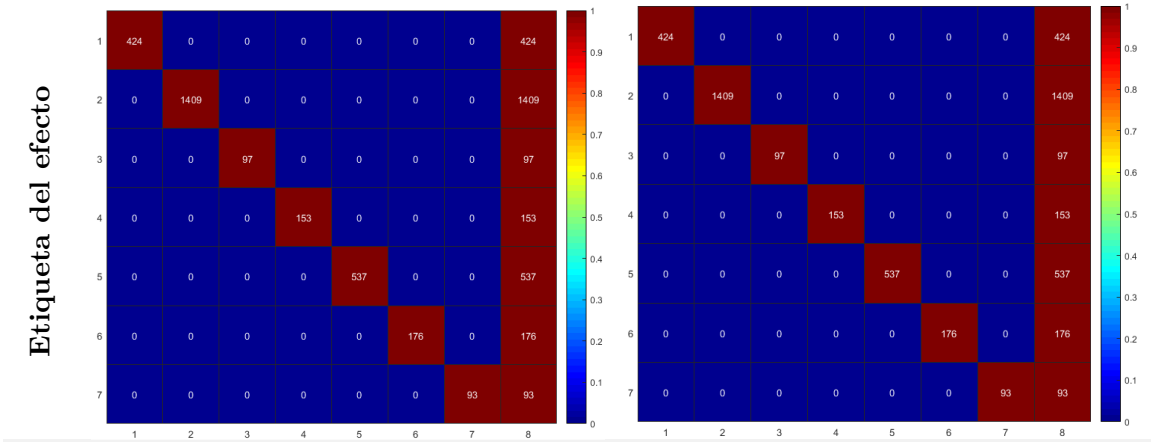


Figura 8.12 Matriz de confusión de efectos CR al inicio, CR al final.

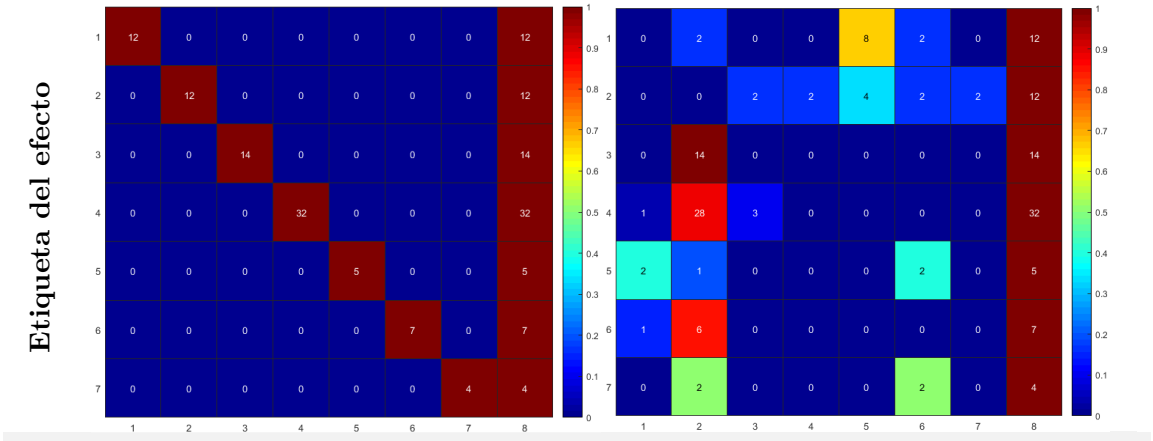


Figura 8.13 Matriz de confusión de efectos CR al inicio e IR al final.

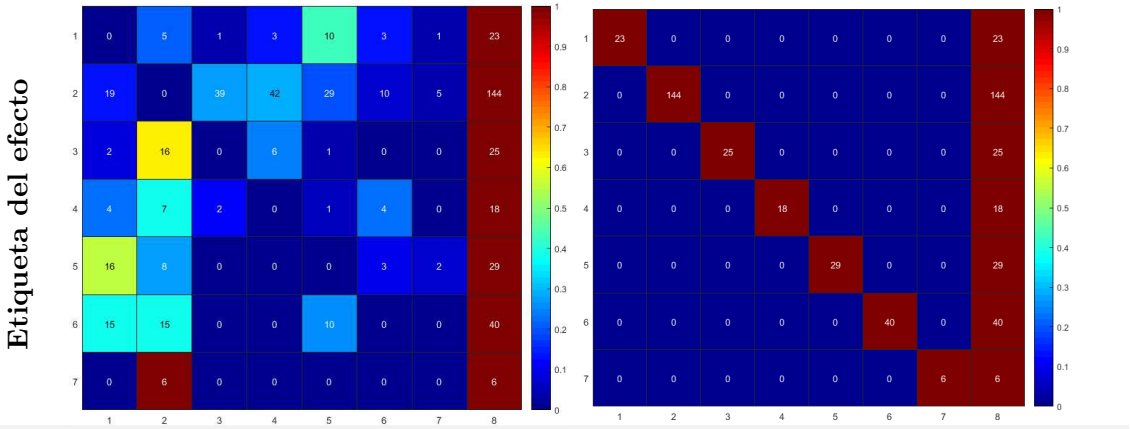


Figura 8.14 Matriz de confusión de efectos IR al inicio, CR al final.

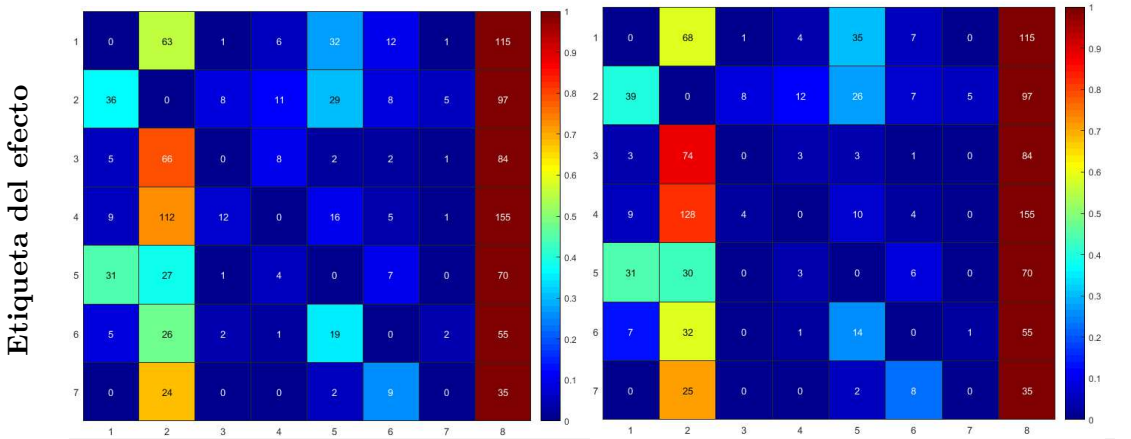


Figura 8.15 Matriz de confusión de efectos IR al inicio, IR al final.