

Prototipo de intérprete de lengua de señas mexicana usando el control Leap Motion

Roberto Hernández-de-la-Luz, Ma. Antonieta Abud Figueroa,
Lisbeth Rodríguez Mazahua, Ulises Juárez Martínez, Celia Romero Torres

Instituto Tecnológico de Orizaba, División de estudios de postgrado e Investigación,
México

{robertohdll,lisbethr}@gmail.com, {mabud,ujuarez,cromero}@ito-depi.edu.mx

Resumen. Actualmente, el 35% de la población mexicana sufre algún tipo de discapacidad auditiva y a pesar de que la lengua de señas mexicana (LSM) se considera una lengua oficial, no se reportan políticas públicas que incentiven el uso y práctica de la lengua, especialmente en los servicios públicos, por lo que las personas con esta discapacidad ven su calidad de vida mermada, debido a que no pueden acceder a servicios como el resto de la población, además de ver limitada su comunicación con todas aquellas personas que no dominan la lengua de señas, además existe un déficit de intérpretes, por lo que muchos organismos públicos y privados encuentran dificultades para implementar planes de capacitación. Teniendo en cuenta esta problemática, este artículo presenta un análisis de tecnologías y una arquitectura de un prototipo de intérprete de lengua de señas mexicana, apoyado en dispositivos de captura de movimiento 3D, con el objetivo de que sirva de base para el desarrollo de un intérprete que brinde una mejora sustancial en la calidad de vida de las personas con discapacidad auditiva, al permitir comunicarse con el resto de la sociedad.

Palabras clave: intérprete, lengua de señas mexicana, sordo, captura de movimiento, leapmotion, LSM, perceptron multicapa, aprendizaje supervisado, aprendizaje automático, SVM.

Prototype of a Mexican Sign Language Interpreter using the Leap Motion Controller

Abstract. Currently, the 35% of the Mexican population suffers from some type of hearing impairment and although the Mexican Sign Language (LSM) is considered an official language, none public policies that encourage the use and practice of the language are reported, especially in public services, this causes that people with this disability see their quality of life degraded, because they cannot access services like the rest of the population, in addition to seeing their

communication limited with all those people who do not know sign language, there is also a deficit of interpreters, so many public and private enterprises find it difficult to implement training plans. Taking into account this problem, this paper presents an analysis of technologies and an architecture of a prototype of a Mexican sign language interpreter, using 3D motion capture devices, in order to serve as a basis for the development of an interpreter that provides a substantial improvement in the quality of life of people with hearing disabilities, by allowing them to communicate with the rest of society.

Keywords: interpreter, Mexican sign language, deaf, motion capture, leap motion, LSM, multilayer perceptron, supervised learning, machine learning, SVM.

1. Introducción

La comunicación por medio de un lenguaje común es una característica inherente de la vida diaria del ser humano, sin embargo, existen personas que sufren diversos tipos de discapacidad que les impide comunicarse, como por ejemplo las personas sordas, lo cual repercute en su calidad de vida.

Por otra parte, en la actualidad existen avances importantes en el desarrollo de dispositivos de captura de movimiento en 3D, además de una constante mejora en la capacidad de los equipos de cómputo, lo cual permite obtener el máximo provecho de diversas bibliotecas de captura y procesamiento de datos en 3D.

Por lo cual, en el presente artículo se presenta un análisis de tecnologías y una arquitectura para un prototipo de intérprete de lengua de señas mexicana por medio de la implementación de un dispositivo de captura de movimiento en 3D para agilizar el procesamiento de la información transmitida, en combinación además de una biblioteca de aprendizaje automático.

Para ofrecer una visión completa de esta investigación, el presente documento se compone de seis secciones, donde la primera sección incluye una breve introducción, la segunda se enfoca en dar a conocer el estado de la práctica, la tercera evalúa las tecnologías existentes, la cuarta describe la arquitectura propuesta, la quinta incluye los resultados obtenidos, la sexta contempla las conclusiones a las que se llegó, finalmente se incluyen las referencias consultadas.

2. Estado de la práctica

En esta sección se dan a conocer algunos trabajos relacionados directa o indirectamente con el artículo presentado.

Leigh et al. [1] realizaron una serie de pruebas para determinar las fortalezas y debilidades del dispositivo “Leap Motion” aplicado en el reconocimiento del lenguaje de señas australiano conocido como “Auslan”, estas pruebas consistieron en evaluar el reconocimiento de la mano y los dedos, en distintas posiciones, así como la capacidad

del dispositivo para identificar correctamente toda la mano al realizar movimientos propios del lenguaje de señas australiano.

Por otra parte, algunos de los problemas que encontraron Leig et al. [1] fueron originados por una API (“Application Programming Interface”, Interfaz de Programación de Aplicaciones) aún incompleta y en etapas tempranas de desarrollo.

Barragan et al. [2] resaltaron la importancia de las características únicas que tiene el lenguaje de señas mexicano, y sobre todo el hecho de que estas sean inherentes al lenguaje propio de México hacen que sea difícil extrapolar una solución ya existente al mismo, por lo que es importante contar con una solución que contemple la estructura gramatical única con la que cuenta.

En el caso de Simos et al. [3], se exploraron las capacidades del dispositivo “Leap Motion” aplicadas al reconocimiento del alfabeto del lenguaje de señas griego, combinando los datos de posicionamiento 3D del dispositivo y usando algoritmos de SVM (“Support Vector Machines”, Máquinas de Vectores de Soporte) para aumentar el porcentaje de clasificación correcto llevándolo sobre el 99%. Dentro de esta misma línea de investigación Mapari et al. [4] realizaron pruebas para verificar la viabilidad del uso del control “Leap Motion” en el reconocimiento de señas del lenguaje de señas americano, concentrándose en el reconocimiento de señas “estáticas”, es decir, el alfabeto y los números del uno al diez, en su caso, obtuvieron una exactitud en la clasificación del 90%.

Existe un amplio interés en el desarrollo de intérpretes de señas, teniendo en cuenta que Sun et al. [5] y Shang et al. [6] propusieron investigaciones utilizando el sensor Kinect y la distorsión en las señas Wifi, en el caso del primero se realizó el experimento utilizando el sensor “Microsoft Kinect” en conjunto con la aplicación de un modelado de LSVM (“Latent Support Vector Machine”, Máquina de Vectores de soporte Latente) para complementar los datos obtenidos por el sensor, es decir, los datos de imágenes 2D, y estructuras tridimensionales capturadas por “Microsoft Kinect” se utilizaron para mejorar la eficiencia en la captura de información relevante, que se usó para apoyar el LSVM. Un dato relevante en [5] es la comprobación de la eficacia de su modelo, para la predicción a nivel de palabras y sentencias, presentando una eficacia por encima del 82% y 84%, respectivamente. En el caso de Shang [6], partiendo de la idea de que los diferentes movimientos de las manos y brazos generan distorsiones únicas en las señales inalámbricas, que a su vez se clasifican como patrones correspondientes con las señas de un lenguaje de señas, bajo el nombre de “WiSign” se presentó el sistema compuesto por tres periféricos, específicamente utilizaron un “router” TP-Link TL-WR1043ND y dos computadoras portátiles Lenovo.

Cabe resaltar la investigación realizada por Bianchi et al. [7] donde señalaron que las personas sordas se comunican esencialmente a través de gestos visuales según el lenguaje de señas que dominan, los cuales tienen una estructura diferente de los lenguajes vocales, por lo que las personas sordas tienen dificultades para aprender y usar las formas escritas de los lenguajes vocales, lo cual limita el acceso a textos y su consiguiente generación, una solución prometedora es “SignWriting”, un marco de trabajo que permite escribir mediante símbolos.

Por otra parte, Rafael et al. [8] esbozaron la idea de una arquitectura de interacción por escenarios para la gente sorda, destacando el hecho de que en México y el resto del

mundo, a pesar de existir legislaciones que buscan promover la integración de las minorías en la sociedad, se necesitan herramientas y propuestas que ayuden a mejorar la inclusión.

Teniendo en cuenta los antecedentes mencionados, el presente trabajo busca validar la eficacia del control Leap Motion para la interpretación de la LSM, teniendo en cuenta las características únicas del lenguaje, así como buscar un conjunto de bibliotecas útiles para el análisis y manipulación de los datos necesarios para el entrenamiento de una red neuronal.

3. Análisis de la tecnología

Esta sección incluye una breve descripción de algunos términos relevantes para la comprensión y desarrollo del tema tratado.

3.1. Dispositivos de hardware disponibles

Después de revisar las publicaciones recientes sobre este tema, se observa que existen múltiples soluciones y enfoques para mejorar la inclusión de las personas, apoyándose en diversos dispositivos de captura de movimiento en 3D, entendiendo que la captura de movimiento, control de movimiento, o “Mocap” (“Motion Capture”, Captura de Movimiento) son términos usados para describir el proceso de grabación de movimiento y la traducción de ese movimiento a un modelo digital [9]. Los principales dispositivos de captura de movimiento en 3D se listan en la tabla 1.

Microsoft Kinect for Xbox One. El sensor de Kinect incluye una cámara RGB para la captura de imágenes en color, además de un sensor que emite ondas infrarrojas junto a otro que permite capturarlas cuando impactan en los objetos, lo cual permite obtener información de profundidad, un micrófono multiarreglo, compuesto por cuatro micrófonos individuales, con lo cual es posible encontrar la ubicación de origen de los sonidos capturados, además de un acelerómetro [10].

Leap Motion. Es un pequeño dispositivo USB (“Universal Serial Bus”, Bus Serial Universal) que contiene tres emisores de luz infrarroja y dos cámaras que capturan las luces infrarrojas de regreso, tiene la capacidad de detectar las palmas de las manos y los movimientos de los dedos; los datos de seguimiento que contienen la posición de ambos, así como la dirección y velocidad son accedidos mediante su SDK (Software Developer Kit); tiene un rango de detección de aproximadamente 0.025m – 0.6m [11].

Myo. La empresa Thalmic desarrolló una banda que se coloca en el brazo y lee la actividad eléctrica de los músculos, cuenta también con acelerómetros que permiten capturar gestos y movimientos de las manos y brazos, se comunica mediante Bluetooth con una computadora para procesar y analizar los gestos generados. Se utiliza para controlar prótesis de brazos en personas amputadas, controlar luces en un escenario y traducir lenguaje de señas americano [12].

Tabla 1. Análisis comparativo de los dispositivos de captura de movimiento en 3D.

Dispositivo	Precio	Método de captura	Rango	Herramientas
Microsoft Kinect for Xbox One	\$2500.00	Receptor infrarrojo, video cámara y audífonos	0.5-4.5 metros	C#, Visual Studio, WPF, Cinder, OpenFrameworks, JavaScript, Vvvv, Processing, Unity3D
Leap Motion	\$1657.00	Receptor infrarrojo	0.025-0.6 metros	JavaScript, Oculus Rift, Unity3D, Unreal
Myo	\$3500.00	Giroscopio y sensores sensibles al tacto	No aplica	Visual Studio
Structure	\$6800.00	Receptor infrarrojo	0.4-3.5 metros	OpenNI, Unity, SceneKit
Intel Real Sense ZR300	\$1962.00	Receptor infrarrojo	0.5-3.5 metros	Java, JavaScript, Processing, Unity3D, Cinder

Structure. El sensor Structure se diseñó para funcionar específicamente con iPads, sin embargo, su SDK y el soporte que tiene para la biblioteca OpenNi 2 permiten que se utilice en otras plataformas como Android, Linux y Windows, dentro de sus características cuenta con un rango de operación de 40 centímetros a 3.5 metros y una precisión de profundidad de 0.5 milímetros, funciona con una conexión USB, cámara VGA (“Video Graphics Array”, Adaptador Gráfico de Video), sensores infrarrojos y batería [13].

Intel Real Sense ZR300. El dispositivo de Intel en su versión ZR300 presenta dos cámaras VGA que permiten tomar fotografías estero, que a su vez se utilizan para determinar la profundidad de los objetos, cuenta con giroscopio y con un rango de operación de mínimo 0.6 metros y un máximo variable según las condiciones de luz existentes, de igual forma utiliza un cable USB 3 [14].

3.2. Algoritmos de aprendizaje

Los algoritmos de aprendizaje se utilizan para predecir las señas ejecutadas por las personas y capturadas por los intérpretes de señas, estos se clasifican en dos grupos dependiendo de la presencia o no de un agente supervisor. Si dicho elemento supervisor está presente durante el aprendizaje, se dice que el aprendizaje es supervisado, en caso contrario es no supervisado [15], siendo relevantes para el presente trabajo la red neuronal *perceptron* multicapa, máquina de vectores de soporte y KNN (*K-Nearest Neighborhood*, K-Vecino más cercano).

4. Solución propuesta

Considerando los costos, ventajas y desventajas de las tecnologías y metodologías analizadas, se determina como solución propuesta el uso del control Leap Motion,

debido a que es el dispositivo que ofrece mayores prestaciones a un menor costo, mostrando además estabilidad en su desarrollo, ya que está en el mercado desde el año 2013 y a lo largo de los años la compañía que lo comercializa ha actualizado constantemente su API, corrigiendo errores y mejorando el rendimiento del producto en cuanto a la detección de distintas posiciones de las manos, por otra parte, durante la investigación del estado de la práctica, se pudo comprobar que se ha utilizado satisfactoriamente en proyectos relacionados con intérpretes de lenguaje de señas.

Para el desarrollo de este trabajo, se utilizaron tres algoritmos, los cuales son: la red neuronal *perceptron* multicapa, máquina de vectores de soporte y *KNN*, en el caso del algoritmo de máquina vectores de soporte, la clasificación es multiclase y se usa un enfoque de clasificación binaria uno contra uno [16], mientras que en el caso del algoritmo *KNN* se seleccionó principalmente por ser uno de los más sencillos e intuitivos, además de ser utilizado regularmente como punto de partida para comparar resultados con otros algoritmos más complejos [17].

4.1. Características relevantes

El control Leap Motion tiene un campo de visión de 150 grados y un rango de efectividad de 0.025 a 0.6 m; además, utiliza un Sistema de coordenadas de 3 ejes, con el origen en el centro del dispositivo, donde el eje Y se encuentra verticalmente con respecto al dispositivo y cuyos valores aumentan positivamente según se aleja del mismo, mientras que el eje Z aumenta los valores positivos según se acerca al usuario, tal como puede apreciarse en la figura 1.

La unidad básica de seguimiento de información que maneja el control Leap Motion es un *frame*, que a su vez se compone de los elementos, mano, dedos y huesos, además de diversos datos correspondientes a la dirección y ángulos de los diferentes segmentos de la mano.

Las características más relevantes tienen relación con la información suministrada por los vectores que indican la posición de los huesos de los dedos con respecto al control Leap Motion, para ello se tomó en cuenta el modelo que maneja el dispositivo y que puede verse en la figura 2, las características fueron:

- El vector con la posición del centro de la palma de la mano (3 características).
- El vector con la dirección de la palma de la mano con respecto a los dedos (3 características).
- El vector con la posición del final de cada hueso de cada dedo (60 características).
- El vector con la dirección de cada dedo (15 características).

Lo cual da un total de 81 características relevantes que son tomadas en cuenta para la alimentación de la red neuronal.

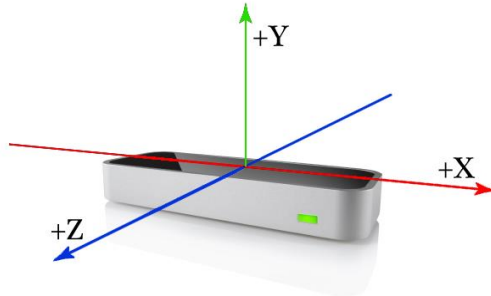


Fig. 1. Sistema de coordenadas de Leap Motion.

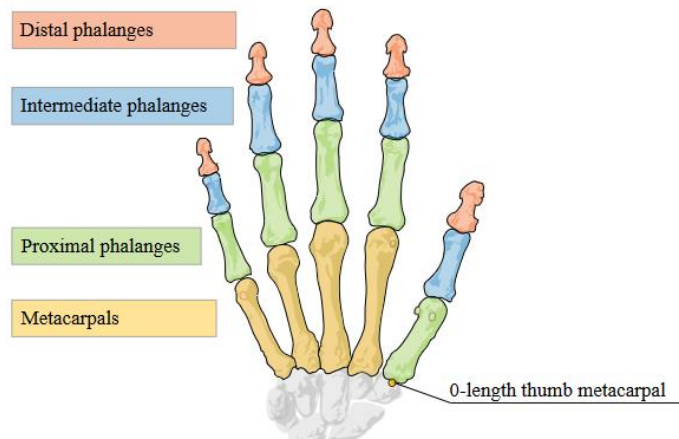


Fig. 2. Modelo de mano manejado por el dispositivo Leap Motion.

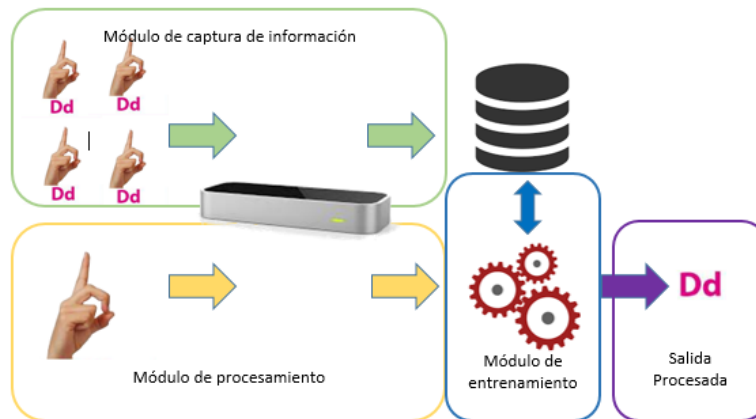


Fig. 3. Módulos propuestos.

Como parte de la optimización de la información se realizó una normalización de los datos correspondientes a los vectores con la posición del final de cada hueso de cada dedo, restándoles el vector de la posición del centro de la palma de la mano, esto permite tener un punto común de referencia, disminuyendo el peso de las características del eje Y, es decir, de esta manera se evita que la posición de la mano con respecto al dispositivo influya negativamente en la clasificación, tal como proponen Simos et al. [3] pero utilizando diferente número de características.

4.2. Arquitectura del intérprete de LSM

Para el prototipo de intérprete de señas mexicano se consideraron cuatro módulos, los cuales se listan a continuación:

1. Módulo de captura de información.
2. Módulo de entrenamiento
3. Módulo de procesamiento
4. Salida Procesada

En la figura 3, se aprecia la distribución y comunicación de los módulos propuestos.

Módulo de captura de información. Este módulo realiza la captura directa de la información correspondiente a la posición y desplazamiento de las manos del usuario, por medio del dispositivo Leap Motion, para después almacenar estos valores en un repositorio de datos.

Módulo de procesamiento. Este módulo realiza la captura de información de la posición y desplazamiento de las manos del usuario, almacenándolas en una estructura en memoria para compararla con el modelo de clasificación seleccionado.

Módulo de entrenamiento. Los datos capturados directamente con el dispositivo Leap Motion y guardados en memoria se utilizan para alimentar al modelo previamente creado y entrenado con los datos guardados en el repositorio, previa optimización de la información recibida para mejorar la clasificación.

Tabla 2. Distribución de muestras por cada letra del alfabeto analizado.

	A	B	C	D	E	F	G	H	I	L	O	S	U	V	W	Y
Mu	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
estr	1	2	3	2	2	2	2	2	3	2	2	2	3	2	2	2
as	4	9	2	9	0	3	9	6	2	6	0	6	0	6	3	6

Tabla 3. Valores promedio obtenidos de los experimentos con los algoritmos de aprendizaje *perceptron* multicapa, KNN y máquina de vectores de soporte.

Algoritmo	Precisión	Velocidad en segundos	Área ROC
KNN	100%	0.023	1
Máquina de Vectores de Soporte	99.86%	0.9999	1
<i>Perceptron</i> Multicapa	99.95%	224.0.3	1



Fig. 4. Posición correspondiente a la letra R y U, en la cual el dispositivo no es capaz de identificar suficientes diferencias.

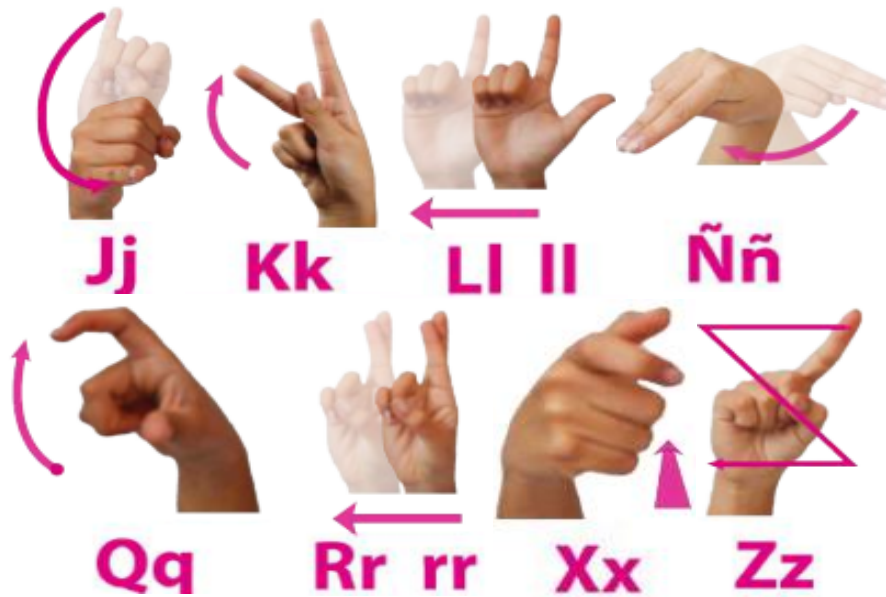


Fig. 5. Posiciones del alfabeto que implican movimiento.



Fig. 6. Posiciones del alfabeto con problemas de identificación debido a la posición de la palma de la mano.

Tabla 4. Matriz de confusión obtenida del clasificador KNN.

	A	B	C	D	E	F	G	H	I	L	O	S	U	V	W	Y
A	111	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0
B	0	131	0	0	0	0	0	0	0	1	0	0	0	0	0	0
C	0	0	123	0	0	1	0	0	0	0	0	0	3	0	2	0
D	0	0	0	125	0	0	0	0	0	0	4	0	0	0	0	0
E	0	0	0	0	120	0	0	0	0	0	0	0	0	0	0	0
F	0	0	0	0	0	123	0	0	0	0	0	0	0	0	0	0
G	0	0	0	0	0	0	129	0	0	0	0	0	0	0	0	0
H	0	0	0	0	0	0	0	126	0	0	0	0	0	0	0	0
I	0	0	0	0	0	0	0	0	132	0	0	0	0	0	0	0
L	0	0	0	0	0	0	0	0	0	126	0	0	0	0	0	0
O	0	0	0	0	0	0	0	0	0	0	120	0	0	0	0	0
S	0	0	0	0	0	0	0	0	0	0	0	126	0	0	0	0
U	0	0	0	0	0	0	0	0	0	0	0	0	130	0	0	0
V	0	0	0	0	0	0	0	0	0	0	0	0	5	121	0	0
W	0	0	0	0	0	0	0	0	0	0	0	0	0	0	123	0
Y	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	126

Tabla 5. Matriz de confusión obtenida del clasificador *perceptron* multicapa.

	A	B	C	D	E	F	G	H	I	L	O	S	U	V	W	Y
A	108	0	0	0	3	0	0	0	0	0	0	3	0	0	0	0
B	0	128	4	0	0	0	0	0	0	0	0	0	0	0	0	0
C	0	0	120	0	0	6	0	0	0	0	0	0	0	0	3	0
D	0	0	0	126	0	0	0	0	0	0	3	0	0	0	0	0
E	0	0	0	0	120	0	0	0	0	0	0	0	0	0	0	0
F	0	0	0	0	0	123	0	0	0	0	0	0	0	0	0	0
G	0	0	0	0	0	0	129	0	0	0	0	0	0	0	0	0
H	0	0	0	0	0	0	1	125	0	0	0	0	0	0	0	0
I	0	0	1	0	0	0	0	0	131	0	0	0	0	0	0	0
L	0	0	0	0	0	0	0	0	0	126	0	0	0	0	0	0
O	0	0	0	2	0	0	0	0	0	0	118	0	0	0	0	0
S	3	0	0	0	0	0	0	0	0	0	0	123	0	0	0	0
U	0	0	0	0	0	0	0	0	3	0	0	0	119	8	0	0
V	0	0	0	0	0	0	0	0	0	0	0	0	13	113	0	0
W	0	0	0	0	0	0	0	0	0	0	0	0	0	0	123	0
Y	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	126

Salida Procesada. Será el significado según el resultado del análisis de los datos obtenidos a través del dispositivo Leap Motion, es decir, el resultado del módulo de entrenamiento.

5. Resultados

El prototipo del sistema propuesto fue implementado en el lenguaje Python, haciendo uso de las bibliotecas externas Pandas para el manejo de estructuras de datos extensas y de manera eficiente, que a su vez obtienen los datos mediante una conexión

a la base de datos manejada mediante la biblioteca SQLAlchemy, finalmente se utilizó la biblioteca de aprendizaje automático Scikit-learn para implementar los algoritmos de clasificación necesarios para comprobar la efectividad de los conjuntos de características seleccionados.

En total se realizaron 2002 registros de 81 características cada uno, correspondientes a 16 letras del alfabeto en la lengua de señas mexicana, en la tabla 2 se pueden observar la cantidad de muestras para cada letra, que corresponden a registros de la mano derecha de una sola persona, el conjunto de datos no está balanceado.

Las letras que no se incluyeron fueron la J, K, LL, M, N, Ñ, P, Q, R, RR, T, X, Z, debido a que algunas implican movimiento y otras presentan problemas para que el dispositivo Leap Motion las identifique correctamente, en la figura 4 se puede observar la letra R, con la cual el dispositivo presenta problemas para identificar la posición de los dedos y la detecta como una letra U, en la figura 5 se identifican las letras que implican movimiento, mientras que en la figura 6 se muestran las posiciones que presentan problemas de identificación debido a la posición de los dedos con la palma de la mano.

En la tabla 3 se observa el promedio de los resultados obtenidos al realizar pruebas con los algoritmos de aprendizaje: *perceptron* multicapa, KNN y máquina de vectores de soporte, aplicando una validación cruzada de 10 pliegues y una repetición de los experimentos veinte veces, a su vez en la tabla 4 se muestra la matriz de confusión del clasificador KNN, en la tabla 5 el resultado de la matriz para *perceptron* multicapa y, por último, en la tabla 6 se incluyen los resultados del clasificador de máquina de vectores de soporte.

Los resultados obtenidos en las matrices de confusión que se muestran en las tablas 4, 5 y 6, aunque varían ligeramente, nos permiten asegurar que para la validación cruzada que se realizó a cada una de ellas, las letras con más problemas son la U y V, seguida de la F y la O, aunque en el caso de la matriz de confusión del *perceptron* multicapa muestra errores en otras letras, estas no se repiten en el resto de algoritmos, por lo que no se consideran relevantes para este primer análisis.

6. Conclusiones

Como se observa en la tabla 2, los tres algoritmos de clasificación tienen una precisión de más del 95% en las predicciones realizadas, lo cual representa un nivel de aceptación alto para las señas estáticas del alfabeto de la lengua de señas mexicana, por lo tanto, se confirma la utilidad y buen desempeño del control Leap Motion y el conjunto de características seleccionado.

De igual manera se identificaron una serie de posiciones únicas e inherentes al vocabulario de la LSM cuyas características presentan problemas para la correcta identificación por parte del control, debido a la posición de los dedos, cuando éstos quedan colocados en medio de otros o se cruzan con la palma de la mano, por otra parte, el manejo de las posiciones que implican una serie de movimientos no se llevó a cabo en este prototipo, pero se tiene contemplado incluirlos en una segunda versión del sistema.

Tabla 6. Matriz de confusión obtenida del clasificador máquina de vectores de soporte.

	A	B	C	D	E	F	G	H	I	L	O	S	U	V	W	Y
A	112	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0
B	0	129	0	0	0	0	0	0	0	3	0	0	0	0	0	0
C	0	0	126	0	0	3	0	0	0	0	0	0	0	0	0	0
D	0	0	0	128	0	0	0	0	0	0	1	0	0	0	0	0
E	0	0	0	0	120	0	0	0	0	0	0	0	0	0	0	0
F	0	0	0	0	0	123	0	0	0	0	0	0	0	0	0	0
G	0	0	0	0	0	129	0	0	0	0	0	0	0	0	0	0
H	0	0	0	0	0	0	126	0	0	0	0	0	0	0	0	0
I	0	0	0	0	0	0	0	132	0	0	0	0	0	0	0	0
L	0	0	0	0	0	0	0	0	126	0	0	0	0	0	0	0
O	0	0	0	0	0	0	0	0	0	120	0	0	0	0	0	0
S	0	0	0	0	0	0	0	0	0	0	126	0	0	0	0	0
U	0	0	0	0	0	0	0	0	0	0	0	130	0	0	0	0
V	0	0	0	0	0	0	0	0	0	0	0	0	126	0	0	0
W	0	0	0	0	0	0	0	0	0	0	0	0	0	123	0	0
Y	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	126

Si bien existen investigaciones anteriores sobre la aplicación del control Leap Motion para la identificación de señas, no se encontró ninguna relacionada específicamente con la LSM que buscara incluir todas las letras del alfabeto, por lo que el presente trabajo permite validar la funcionalidad y efectividad de la combinación del control y los algoritmos de aprendizaje supervisado.

Como trabajo futuro se tiene planificado investigar las opciones disponibles para realizar el análisis de las letras y palabras que implican una secuencia de posiciones o movimientos, además de buscar posibles optimizaciones a la selección de características relevantes, aplicando el resultado obtenido a un caso de estudio seleccionado que permita delimitar el vocabulario disponible para mejorar el porcentaje de exactitud en la identificación de letras o palabras.

Agradecimientos. Los autores agradecen al Tecnológico Nacional de México y al Consejo Nacional de Ciencia y Tecnología (CONACYT) por el patrocinio brindado para la realización de este trabajo.

Referencias

1. Potter, L.E., Araullo, J., Carter, L.: The Leap Motion controller: a view on sign language. In: Proceedings of the 25th Australian Computer-Human Interaction Conference: Augmentation, Application, Innovation, Collaboration, pp. 175–178 (2013)
2. Barragán, J., Javier, F., Pérez-Grana, J.A., Cervantes, F., Morris, S.K., Olide-Márquez, M.G., Pérez-Sánchez, A.P.: Spanish sign language interpreter for Mexican linguistics. J. Comput. Sci. Technologies 13, pp. 32–37 (2013)
3. Simos, M., Nikolaidis, N.: Greek sign language alphabet recognition using the leap motion device. In: Proceedings of the 9th Hellenic Conference on Artificial Intelligence, pp. 1–4 (2016)

4. Mapari, R.B., Kharat, G.: American Static Signs Recognition Using Leap Motion Sensor. In: Proceedings of the Second International Conference on Information and Communication Technology for Competitive Strategies, pp. 1–5 (2016)
5. Tianzhu, S.C.: Latent Support Vector Machine Modeling for Sign Language Recognition with Kinect. *J. (ACM) Trans. Intell. Syst. Technol, TIST*. VI, pp. 1–20 (2015)
6. Shang, J., Wu, J.: A Robust Sign Language Recognition System with Multiple Wi-Fi Devices. In: Proceedings of the Workshop on Mobility in the Evolving Internet Architecture, pp. 19–24 (2017)
7. Bianchini, C.S., Borgia, F., Bottoni, P., Marsico, M.D.: SWift: a SignWriting improved fast transcriber. In: Proceedings of the International Working Conference on Advanced Visual Interfaces, pp. 390–393 (2012)
8. Rojano-Cáceres, J.R., Sánchez-Barrera, H., Martínez-Gutiérrez, M.E., Molero-Castillo, G., Ortega-Carrillo, J.A.: Designing an interaction architecture by scenarios for Deaf people. In: Proceedings of the XVII International Conference on Human Computer Interaction, pp. 1–2 (2016)
9. Crespo, M. A.: Dirección Cinematográfica: Manual Avanzado de Aprendizaje Creativo (2013)
10. MICROSOFT: Kinect for Windows Sensor Components and Specifications, <https://msdn.microsoft.com/en-us/library/jj131033.aspx> (2017)
11. Shao, L.: Hand movement and gesture recognition using Leap Motion Controller (2016)
12. THALMIC: <https://www.thalmic.com/> (2018)
13. STRUCTURE.IO: Precise 3D vision for embedded applications. <https://structure.io/embedded> (2018)
14. Intel® RealSense™: Development Kit Featuring the ZR300, <https://click.intel.com/intel-realsensetm-development-kit-featuring-the-zr300.html> (2018)
15. Lahoz-Beltrá, R.: Bioinformática: simulación, vida artificial e inteligencia artificial. Diaz de Santos (2004)
16. Steinwart, I., Christmann, A.: Support Vector Machines. Springer, New York (2008)
17. Rajaguru, H.; Prabhakar, S.K.: KNN Classifier and K-Means Clustering for Robust Classification of Epilepsy from EEG Signals. A Detailed Analysis. Anchor Academic Publishing (2017)