

REDUCCIÓN DE LA DIMENSIONALIDAD DE UN SISTEMA DE PARTÍCULAS EN MOVIMIENTO UTILIZANDO REDES NEURONALES CONVOLUCIONALES

DIMENSIONALITY REDUCTION OF A MOVING PARTICLE SYSTEM USING CONVOLUTIONAL NEURAL NETWORKS

Sergio E. Bertone, Gabriel D. Puccini, Melina Denardi, Carlos A. Bonetti y Jezabel D. Bianchotti

Laboratorio de Métodos y Simulaciones Computacionales, Facultad Regional Rafaela, Universidad Tecnológica Nacional, Acuña 49, Rafaela 2300, Argentina, gabriel.puccini@frra.utn.edu.ar (G.D.P)

Palabras clave: Autoencoders, Redes convolucionales, Reducción dimensional, Materia granular

Resumen. En el campo del aprendizaje automático, en los últimos años se ha desarrollado un enfoque innovador para modelar sistemas complejos mediante la identificación de dimensiones intrínsecas y variables de estado neuronales (NSV) utilizando redes neuronales convolucionales (CNN). Esta técnica ha demostrado ser eficaz en la predicción estable a largo plazo de sistemas dinámicos complejos, como los encontrados en aplicaciones industriales de flujos de materia granular. El presente trabajo se centra en el uso de autoencoders convolucionales para predecir imágenes en sistemas bidimensionales de partículas en movimiento. Los datos, generados mediante simulaciones, se utilizan para entrenar la red, descomponiendo y reconstruyendo secuencias de video que capturan la dinámica del sistema. El proceso de descomposición implica una reducción de la dimensionalidad del sistema, lo que es crucial para obtener una representación más eficiente y simplificada de su comportamiento dinámico. Esto facilita la comprensión y predicción del sistema en entornos industriales. Se espera que los resultados obtenidos contribuyan al desarrollo de futuros modelos simplificados del sistema.

Keywords: Autoencoders, Convolutional Networks, Dimensionality Reduction, Granular Matter

Abstract. In the field of machine learning, an innovative approach has been developed in recent years for modeling complex systems by identifying intrinsic dimensions and neural state variables (NSVs) using convolutional neural networks (CNNs). This technique has proven effective for long-term stable prediction of complex dynamic systems, such as those encountered in industrial applications involving granular matter flows. The present work focuses on the use of convolutional autoencoders to predict images in two-dimensional particle systems in motion. The data, generated through simulations, are used to train the network, decomposing and reconstructing video sequences that capture the system's dynamics. The decomposition process involves a reduction of the system's dimensionality, which is crucial for obtaining a more efficient and simplified representation of its dynamic behavior. This facilitates the understanding and prediction of the system in industrial environments. The results are expected to contribute to the development of future simplified models of the system.

1. INTRODUCCIÓN

Los flujos granulares son fenómenos complejos que se manifiestan en procesos naturales y en diversas aplicaciones industriales, donde las partículas sólidas interactúan y muestran un comportamiento colectivo. Estos flujos son particularmente relevantes en operaciones como la descarga de materiales granulares en tolvas, donde la dinámica de las partículas está influenciada por múltiples factores, como las interacciones entre las partículas, y entre éstas y las paredes del silo (Cundall y Strack, 1979; Nedderman, 1992). La modelización de estos sistemas presenta un desafío significativo dado que la naturaleza discreta y no lineal de las interacciones dificulta su representación mediante modelos continuos convencionales.

Para abordar esta complejidad en los últimos años se ha consolidado el uso de los Métodos de Elementos Discretos (DEMs), que permite simular el comportamiento de un ensamble de partículas mediante la dinámica de las partículas individuales y sus interacciones. Este enfoque considera un número finito de partículas que interactúan a través de fuerzas de contacto y no contacto, describiendo su movimiento de acuerdo con las leyes del movimiento de Newton (Cundall y Strack, 1979; Zhu et al., 2007, 2008). Sin embargo, las simulaciones DEM requieren un considerable esfuerzo computacional lo que limita su aplicabilidad en escenarios que requieren un gran número de simulaciones para la calibración de los parámetros del modelo.

En este contexto, la reducción de dimensionalidad emerge como una herramienta crucial para simplificar modelos discretos de un número elevado de partículas. Históricamente, esta reducción se ha abordado mediante técnicas como el Análisis de Componentes Principales (PCA), que proyecta datos de alta dimensionalidad en un subespacio lineal de menor dimensión (Pearson, 1901; Holmes et al., 2012). La Descomposición en Valores Singulares (SVD) también ofrece una metodología sistemática para obtener una aproximación de baja dimensionalidad a partir de datos complejos, apoyándose en la técnica descomposición ortogonal (POD) (Berkooz et al., 1993; Taira et al., 2017). No obstante, aunque SVD es eficaz para sistemas dinámicos basados en observaciones, su capacidad para procesar imágenes de video y extraer variables de estado sin información adicional es limitada, lo cual podría ser fundamental para una comprensión más profunda de la física en fenómenos complejos (Chen et al., 2021).

A pesar de la utilidad de estas técnicas para simplificar datos y mejorar la interpretabilidad de los modelos, a menudo se enfrentan a limitaciones cuando se trata de relaciones no lineales, especialmente en el contexto de flujos granulares. Las extensiones no lineales del PCA han sido facilitadas por el uso de redes neuronales, conocidas como autoencoders (Baldi y Hornik, 1989; Goodfellow et al., 2016). Estas arquitecturas permiten capturar estructuras complejas y no lineales en los datos, ofreciendo una perspectiva más adecuada para modelar fenómenos complejos que las técnicas lineales tradicionales.

Aunque numerosas herramientas de aprendizaje automático han demostrado ser efectivas para modelar la dinámica de sistemas físicos, muchas de estas técnicas requieren que se proporcionen previamente mediciones específicas de las variables de estado. Por ejemplo, el trabajo de Brunton et al. (Brunton et al., 2016) requirió las coordenadas espaciales y sus derivadas para modelar el sistema de Lorenz. Además, Champion et al. (Champion et al., 2019) utilizaron funciones base predefinidas para guiar el entrenamiento de un autoencoder en la reconstrucción de observaciones. Otros autores combinaron redes neuronales con propiedades físicas conocidas para resolver ecuaciones, partiendo de variables previamente definidas (Udrescu y Tegmark, 2020). Estos enfoques subrayan la dependencia significativa de datos explícitos, lo que puede limitar la aplicabilidad de los métodos en escenarios donde tales datos no están disponibles o son difíciles de obtener.

Como un estudio preliminar en la reducción dimensional de un sistema granular de alta dimension, en este trabajo se utilizan autoencoders para descomponer y reconstruir imágenes de video de un sistema de tres partículas interactuantes. La metodología empleada incluye la generación de secuencias de video a partir de simulaciones computacionales del problema de tres cuerpos bidimensional. Estas imágenes son posteriormente utilizadas para entrenar dos modelos de autoencoder, uno de los cuales logra capturar las características más relevantes del sistema y predecir con precisión el comportamiento dinámico en los cuadros siguientes.

2. METODOLOGÍA

2.1. Datos para entrenamiento y prueba de las redes neuronales.

El conjunto de datos empleado fue generado computacionalmente, simulando el movimiento de tres partículas circulares en un dominio bidimensional. Durante el transcurso de la simulación las partículas interactúan entre sí y con los bordes del dominio, modificando sus trayectorias y velocidades. De este modo se simula la filmación de un sistema dinámico, donde cada imagen generada corresponde a un cuadro en una secuencia temporal. Cada simulación tuvo una duración de 2 segundos, de este modo se obtuvieron 120 imágenes de 128 x 128 píxeles en escala de grises para cada simulación, para una frecuencia de 60 cuadros por segundo. Se realizaron 150 simulaciones, adoptando posiciones, velocidades y direcciones iniciales aleatorias para las partículas.

Para representar el comportamiento del sistema dinámico, las imágenes se agruparon en secuencias de cuatro cuadros consecutivos. Los dos primeros cuadros (capturando los estados del sistema en los tiempos t_i y t_{i+1}) se utilizaron como datos de entrada para los modelos, mientras que los dos cuadros siguientes (en los tiempos t_{i+2} y t_{i+3}) se usaron como datos de salida. De este modo, para cada simulación se obtuvieron 117 secuencias. Las dos imágenes de entrada y las dos de salida se concatenaron verticalmente, formando imágenes de 256x128 píxeles que agrupan dos cuadros consecutivos. Finalmente, las imágenes concatenadas se dividieron en tres conjuntos: entrenamiento (14040 imágenes obtenidas de 120 simulaciones), evaluación (1755 imágenes de 15 simulaciones) y prueba (1755 imágenes de 15 simulaciones).

Todas las imágenes del conjunto de datos fueron normalizadas a un rango de valores entre 0 y 1 antes de ser procesadas por los modelos. Esta normalización tiene como objetivo facilitar el proceso de entrenamiento al reducir la variabilidad en los valores de los píxeles, lo que ayuda a que la red converja más rápidamente. Al mantener los valores dentro de un rango fijo, se previene que las diferencias de escala entre las entradas afecten el rendimiento del modelo, promoviendo un aprendizaje más estable y eficiente. Además, esta práctica es común en el preprocesamiento de datos, ya que permite que las funciones de activación, como ReLU, operen de manera más efectiva, mejorando así el rendimiento general del modelo.

2.2. Arquitectura de los autoencoders

En la Figura 1 se muestra el esquema de la arquitectura del autoencoder con las imágenes concatenadas de entrada y de salida. El encoder recibe dos imágenes de entrada concatenadas y las transforma progresivamente a través de las capas convolucionales y de pooling, reduciendo su dimensión espacial mientras incrementa la complejidad de las características extraídas. Este espacio de representación comprimido es luego utilizado por el decodificador para reconstruir las dos imágenes siguientes en la secuencia temporal.

Se propusieron dos modelos de autoencoder, que denominaremos *modelo simple* y *modelo profundo*.

- Modelo simple: consiste en un codificador de tres bloques de capas convolucionales y de pooling, seguido de un decodificador simétrico.
- Modelo profundo: compuesto por un codificador con siete bloques de capas convolucionales y de pooling, con una capa densa en el cuello de botella, seguido por un decodificador que invierte los bloques del codificador.

Ambas arquitecturas se implementaron utilizando el entorno de trabajo para aprendizaje profundo Keras (Chollet, 2015) bajo el modelo funcional, lo que permitió flexibilidad para definir y conectar las diversas capas.

En los modelos de autoencoder utilizados, las imágenes de entrada y salida, con una resolución de 256x128 píxeles (32,768 dimensiones), se codifican en un espacio latente de menor dimensionalidad, con el objetivo de lograr que el modelo capture las características más relevantes del sistema dinámico. En el modelo simple, el cuello de botella tiene una dimensionalidad de 32x16 píxeles con 8 canales, lo que equivale a un espacio latente de 4,096 dimensiones. Por otro lado, en el modelo profundo, la dimensionalidad del cuello de botella es considerablemente menor: 64 neuronas que representan un vector de 64 dimensiones. Estas neuronas salen de 64 canales tras la última convolución antes del cuello de botella, codificando las características del sistema de forma aún más comprimida.

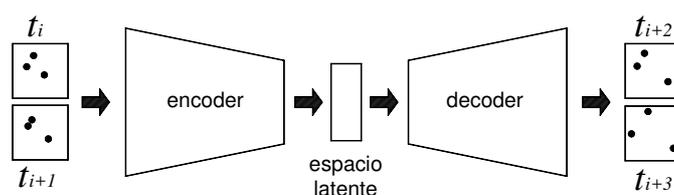


Figura 1: Esquema del autoencoder.

2.3. Entrenamiento y evaluación de las redes neuronales

Ambos modelos fueron entrenados usando el entorno de trabajo Keras, bajo la API funcional, con el optimizador Adam y la función de pérdida binary crossentropy. Se realizaron experimentos entrenando ambos modelos por 20 y 50 épocas, y los datos de evaluación se utilizaron para monitorear el rendimiento en términos de la capacidad de predicción de los frames de salida.

Se utilizó el optimizador ADAM (Adaptive Moment Estimation), utilizado en los autoencoders convolucionales por su capacidad para ajustar dinámicamente las tasas de aprendizaje, lo que permite una convergencia más rápida y eficiente, y por su eficiencia en problemas con grandes volúmenes de datos y parámetros.

La función de pérdida utilizada fue Binary Crossentropy, utilizada comúnmente en la reconstrucción de imágenes en escala de grises. Dado que las imágenes están normalizadas entre 0 y 1, esta función de pérdida es adecuada para comparar la similitud entre los valores de los píxeles de las imágenes predichas y las imágenes reales. La elección de la entropía cruzada también se justifica por su capacidad para penalizar de manera más severa las predicciones incorrectas en los píxeles, lo que resulta en una mejor reconstrucción visual de las imágenes.

Para realizar una evaluación cuantitativa de la calidad de las imágenes generadas por los modelos, se empleó el Índice de Similitud Estructural (SSIM, Structural Similarity Index). Este índice mide la similitud entre dos imágenes considerando aspectos como la luminancia, el contraste y la estructura.

El SSIM se define como (Wang, 2004):

$$\text{SSIM}(x, y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)} \quad (1)$$

donde

- μ_x y μ_y son las medias de las imágenes x e y , respectivamente.
- σ_x^2 y σ_y^2 son las varianzas de las imágenes x e y .
- σ_{xy} es la covarianza entre las imágenes x e y .
- C_1 y C_2 son constantes pequeñas que estabilizan la división.

3. RESULTADOS

A continuación se presentan los resultados obtenidos al comparar los dos modelos de autoencoder propuestos: el *modelo simple* (compuesto por tres bloques de capas convolucionales y de pooling) y el *modelo profundo* (de siete bloques de capas convolucionales y de pooling, con capa densa en el cuello de botella). Ambos modelos fueron entrenados para simular el movimiento de tres partículas dentro de un espacio bidimensional.

A lo largo de esta sección, el término *entradas* hará referencia a las dos primeras imágenes consecutivas de cada secuencia de cuatro cuadros en que se ha dividido la animación (frames 1 y 2, para los tiempos t_i y t_{i+1}). Por su parte, el término *salidas reales* corresponderá a las imágenes de los dos cuadros siguientes de dicha secuencia (frames 3 y 4, correspondientes a los tiempos t_{i+2} y t_{i+3}). El término *salidas predichas* hará referencia a las imágenes generadas por los dos modelos de autoencoder a partir de las *entradas*, las que se compararán con las *salidas reales* para evaluar las capacidades de predicción de cada modelo.

3.1. Comparación cualitativa.

En la Fig. 2 se muestran ejemplos de las salidas predichas por los modelos a partir de datos a los que no han tenido acceso durante su entrenamiento, es decir, pertenecientes al conjunto de prueba, junto con las entradas y salidas reales correspondientes a cada ejemplo. Las imágenes mostradas corresponden a modelos entrenados durante 20 y 50 épocas.

El modelo simple genera imágenes que aparecen borrosas y que muestran mayor similitud con las entradas que con las salidas reales, comportándose de un modo similar a un autoencoder clásico que reconstruye las imágenes de entrada.

El modelo profundo produce imágenes más definidas y semejantes a las salidas reales, lo que indica su capacidad para capturar la variación temporal y espacial del sistema, mostrando de este modo una mejora significativa en la calidad de las predicciones respecto al modelo simple. Las formas de las imágenes generadas se vuelven más precisas a medida que aumenta el número de épocas de entrenamiento; por ejemplo, se observa que las partículas obtenidas con el entrenamiento durante 20 épocas tienden a aparecer deformadas, lo que indica que el modelo está aún en proceso de ajuste y refinamiento.

En la Fig. 3 se presenta una comparación entre los cuadros reales y los predichos por el modelo profundo en cinco instantes de tiempo, con intervalos de 0.5 segundos, correspondientes a una de las simulaciones del conjunto de prueba. Esta comparación resalta cómo el modelo profundo logra capturar la dinámica del sistema y su evolución a lo largo del tiempo.

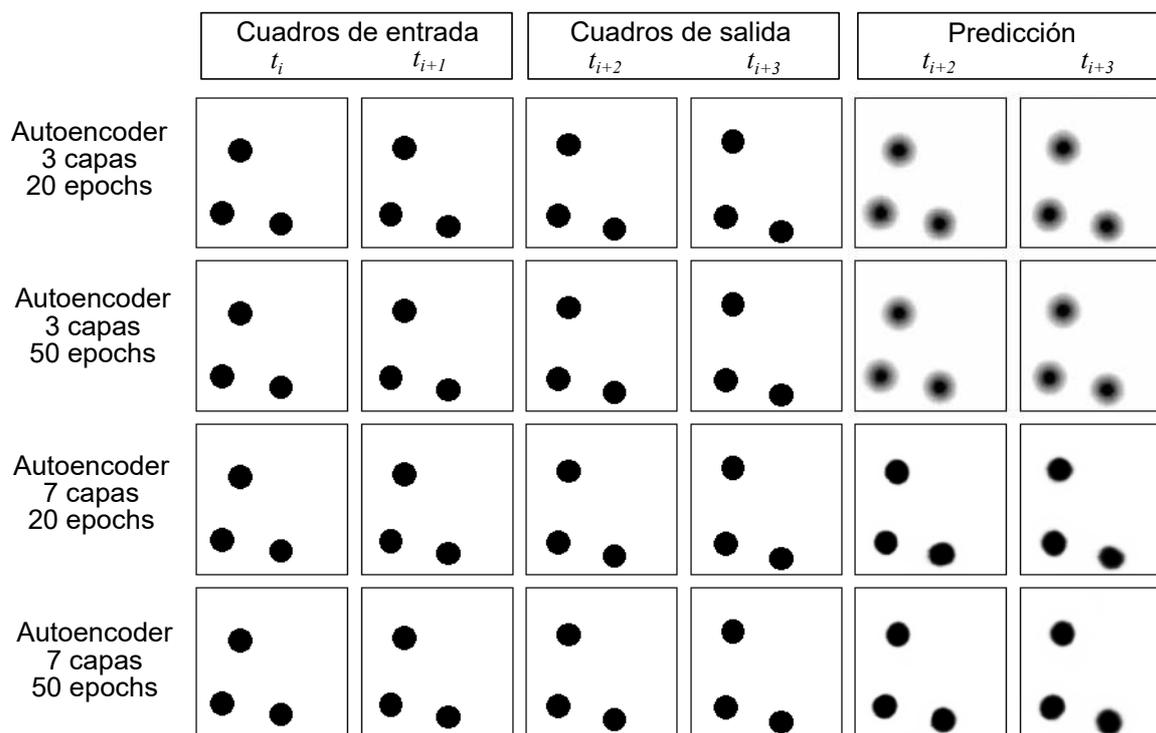


Figura 2: Comparación de entradas y salidas para los dos modelos de encoder.

3.2. Comparación cuantitativa. Métrica de similitud estructural (SSIM).

Para evaluar cuantitativamente la calidad de las salidas generadas por los modelos se compararon entre sí los tres tipos de imágenes (entradas, salidas reales y salidas predichas) del conjunto de datos de prueba. La comparación se basó en el índice SSIM, utilizando 1755 pares de imágenes concatenadas verticalmente (cuadros consecutivos) de cada tipo. La tabla 1 muestra el promedio de los índices calculados para las distintas combinaciones, discriminados por modelo utilizado y cantidad de épocas de entrenamiento.

Para el modelo simple, el promedio de índices SSIM entre las salidas predichas y las salidas reales (0,86) es ligeramente más bajo que entre las salidas predichas y las entradas (0,88). Esto indica que las imágenes generadas por el modelo simple son más similares a las entradas que

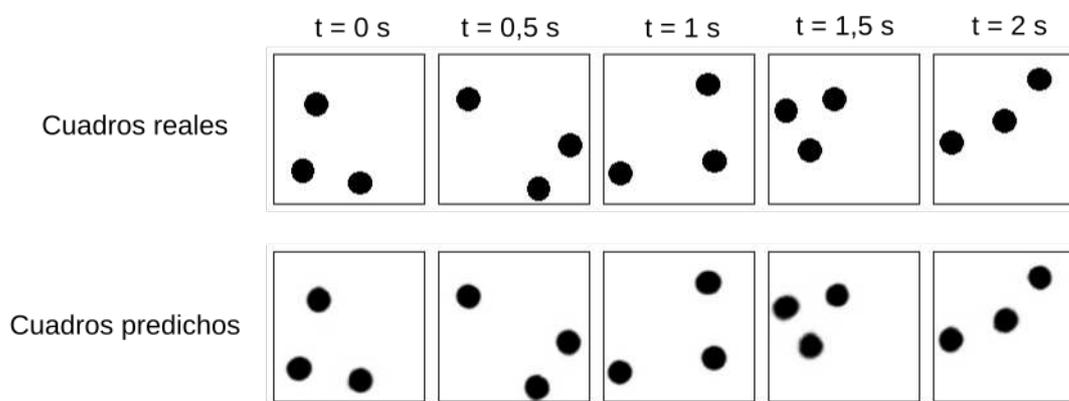


Figura 3: Comparación de la evolución temporal entre cuadros reales y predichos por el modelo profundo.

pares comparados	modelo simple		modelo profundo	
	20 epochs	50 epochs	20 epochs	50 epochs
salidas predichas - salidas reales	0,863	0,864	0,929	0,932
salidas predichas - entradas	0,874	0,876	0,898	0,899
salidas reales - entradas	0,901	0,901	0,901	0,901

Tabla 1: Promedio de índices SSIM para el conjunto de imágenes de prueba.

a las salidas reales, lo que sugiere que este modelo tiende a replicar las entradas en lugar de capturar las características dinámicas del sistema.

En contraste, el modelo profundo muestra un rendimiento superior. En este caso el promedio de índices SSIM entre salidas predichas y salidas reales alcanza un valor considerablemente alto (0,93), lo que sugiere una alta similitud estructural entre las imágenes correspondientes a las salidas predichas y a las salidas reales. Estos valores corresponden a secuencias temporales extraídas de 15 simulaciones del conjunto de prueba, cada una compuesta por 120 cuadros, lo que refleja la capacidad del modelo profundo para capturar de modo efectivo la evolución dinámica del sistema simulado a lo largo del tiempo. Además, la similitud entre los promedios SSIM para los pares de imágenes “salidas predichas - entradas” y “salidas reales - entradas” también confirma este comportamiento del autoencoder.

La Fig. 4 muestra un ejemplo de la comparación entre las imágenes (cuadros de video consecutivos) correspondientes a las salidas reales y las salidas predichas para ambos modelos entrenados durante 50 épocas. En la fila A se presenta una imagen de salida real, su imagen de salida predicha y el mapa SSIM de la comparación entre ambas para el modelo profundo. El mapa SSIM muestra una alta similitud entre las salidas predichas y las salidas reales, evidenciada por la predominancia de colores blancos en las regiones centrales de las partículas. Esto indica que el modelo profundo logra capturar con precisión la forma de las partículas y sus posiciones. Por otro lado, en la fila B correspondiente al modelo simple, el mapa SSIM revela una baja similitud entre las imágenes, con áreas de color negro cerca del centro de las partículas y zonas grises y negras deformadas alrededor de lo que deberían ser las partículas. Ambos mapas SSIM se generaron a partir de la misma imagen de entrada, permitiendo de este modo una comparación directa del rendimiento de los dos modelos en cuanto a la calidad de sus predicciones.

4. CONCLUSIONES

La evaluación de los dos modelos de autoencoder muestra que el modelo profundo supera al modelo simple en términos de predicción del movimiento de partículas en un espacio bidimensional. El modelo profundo, con su arquitectura más compleja, produce imágenes que se asemejan mucho más a las salidas reales que las generadas por el modelo simple. Esto se refleja en el alto índice SSIM del modelo profundo, que indica una capacidad superior para capturar tanto la variación temporal como espacial del sistema.

Por otro lado, el modelo simple muestra una tendencia a replicar las imágenes de entrada en lugar de predecir las características dinámicas del sistema, como se observa en su menor índice SSIM en comparación con el modelo profundo. La calidad de las predicciones del modelo profundo mejora significativamente con un mayor número de épocas de entrenamiento, confirmando que una mayor complejidad del modelo y más tiempo de entrenamiento contribuyen a una mejor capacidad predictiva.

Se debe tener en cuenta que en el modelo simple, el cuello de botella no es un vector sino una

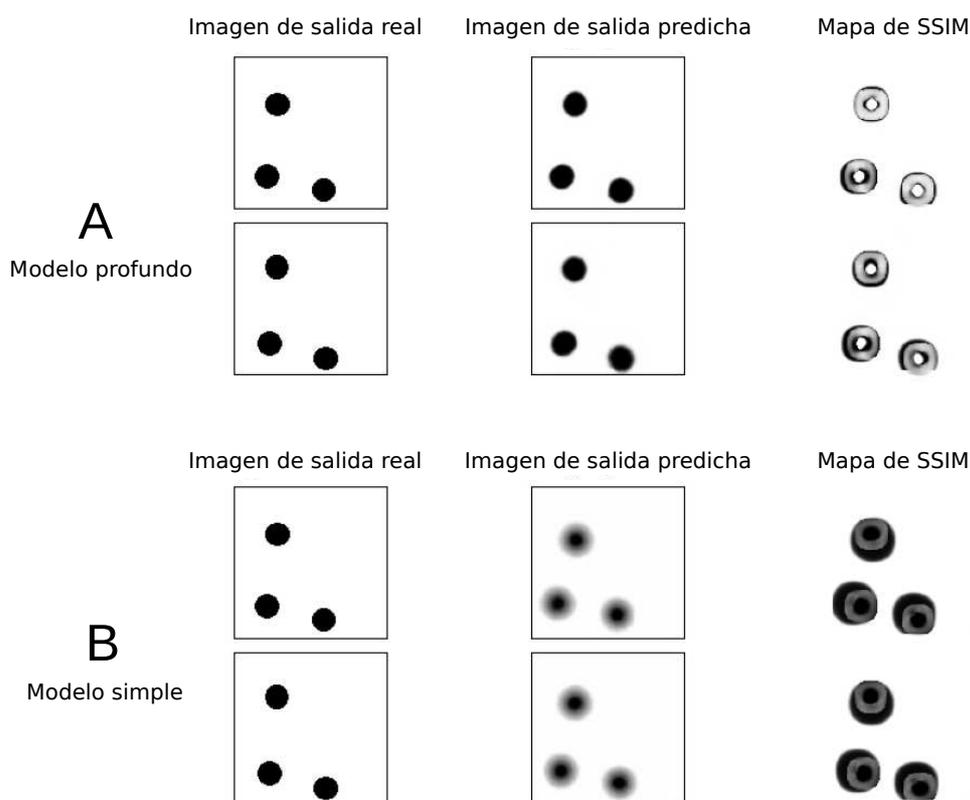


Figura 4: Ejemplo de comparación de métrica SSIM.

representación de múltiples canales. Por otro lado, en el modelo profundo el espacio latente está estructurado como un vector, lo que facilita la extracción y análisis de las variables latentes más relevantes. Estas variables latentes capturan la estructura intrínseca del sistema, permitiendo la reconstrucción de las imágenes de salida y la predicción de estados futuros del sistema dinámico. Esta capacidad de análisis abre la posibilidad de investigar con mayor profundidad las propiedades de las variables latentes, así como su aplicación en otros sistemas dinámicos complejos, lo que podría conducir a modelos más eficientes y precisos en la predicción y simulación de sistemas físicos.

Por último, se espera que los resultados de este trabajo contribuyan de manera significativa al campo de la investigación en flujos granulares, proporcionando un marco para el desarrollo de modelos simplificados que puedan aplicarse en la práctica industrial. Al integrar simulaciones de sistemas dinámicos con técnicas de aprendizaje automático, este estudio ofrece una perspectiva innovadora para abordar la complejidad de los sistemas granulares, mejorando así la capacidad de predicción y control en entornos industriales.

AGRADECIMIENTOS

Los autores agradecen a la Universidad Tecnológica Nacional por financiar esta investigación en el marco del proyecto con Código ASPPRA0010245.

REFERENCIAS

- Baldi P. y Hornik K. Neural networks and principal component analysis: Learning from examples without local minima. *Neural Networks*, 2(1):53–58, 1989.
- Berkooz G., Holmes P., y Lumley J.L. The proper orthogonal decomposition in the analysis of turbulent flows. *Annual Review of Fluid Mechanics*, 25(1):539–575, 1993.
- Brunton S.L., Proctor J.L., y Kutz J.N. Proceedings of the national academy of sciences. *Proceedings of the National Academy of Sciences*, 113(14):3932–3939, 2016. doi:10.1073/pnas.1517792113.
- Champion K., Lusch B., Kutz J.N., y Brunton S.L. Proceedings of the national academy of sciences. *Proceedings of the National Academy of Sciences*, 116(45):22445–22454, 2019. doi:10.1073/pnas.1911627116.
- Chen B., Huang K., Raghupathi S., Chandratreya I., Du Q., y Lipson H. Discovering state variables hidden in experimental data. 2021. doi:10.48550/arXiv.2112.10755.
- Chollet F. Keras. *GitHub repository*, 2015.
- Cundall P. y Strack O. A discrete numerical model for granular assemblies. *Geotechnique*, 29:47–65, 1979.
- Goodfellow I., Bengio Y., Courville A., y Bengio Y. *Deep Learning*, volumen 1. MIT Press, Cambridge, 2016.
- Holmes P.J., Lumley J.L., Berkooz G., y Rowley C.W. *Turbulence, coherent structures, dynamical systems and symmetry*. Cambridge Monographs in Mechanics. Cambridge University Press, Cambridge, England, 2nd edición, 2012.
- Nedderman R. Statics and kinematics of granular materials. *Cambridge University Press, London*, 1992.
- Pearson K. On lines and planes of closest fit to systems of points in space. *Philosophical Magazine*, 2(7-12):559–572, 1901.
- Taira K., Brunton S.L., Dawson S.T.M., Rowley C.W., Colonius T., McKeon B.J., Schmidt O.T., Gordeyev S., Theofilis V., y Ukeiley L.S. Modal analysis of fluid flows: An overview. *AIAA Journal*, 55(12):4013–4041, 2017.
- Udrescu S.M. y Tegmark M. Science advances. *Science Advances*, 6(23):eaay2631, 2020. doi:10.1126/sciadv.aay2631.
- Wang Z. Image quality assessment: Form error visibility to structural similarity. *IEEE Trans. Image Process.*, 13(4):604–606, 2004.
- Zhu H., Zhou Z., Yang R., y Yu A. Discrete particle simulation of particulate systems: theoretical developments. *Chemical Engineering Science*, 62(13):3378–3396, 2007.
- Zhu H., Zhou Z., Yang R., y Yu A. Discrete particle simulation of particulate systems: a review of major applications and findings. *Chemical Engineering Science*, 63(23):5728–5770, 2008.