

GANdela

Alfredo Montero López
Leonardo Amaro Rodríguez
Anthuan Montes de Oca
Yisell Martínez Noa

Índice

1. Introducción	2
1.1. Motivación	2
1.2. Problemática	2
1.3. Objetivos Generales	2
1.4. Objetivos Específicos	2
2. Estado del Arte / Preliminares	3
2.1. Redes Generativas Adversarias (GANs)	3
2.2. Pix2Pix GAN	3
2.3. Aplicaciones de Pix2Pix GAN y CycleGan en Cartografía	4
2.4. Desafíos y Limitaciones	4
3. Propuesta de Solución	4
3.1. ¿Qué es pix2pix-GAN?	4
4. Base de Datos	5
5. Resultados de las pruebas	6
5.1. Intentos de configuración	6
5.1.1. L1 con BCEwithLogits	6
5.1.2. L1 with MSE	6
5.1.3. MSE with BCEwithLogits	6
5.2. Modelo final	7
5.2.1. MSE con MSE con dos tiempos	7
5.2.2. Resultados	7
6. Discusión de los resultados	8
6.1. Repercusión ética de las soluciones	8
6.1.1. Privacidad y Seguridad	8
6.1.2. Transparencia y Explicabilidad	8
6.1.3. Equidad y Accesibilidad	9

6.1.4. Impacto Ambiental	9
6.1.5. Uso Responsable de la Tecnología	9
7. Conclusiones y trabajo futuro	9
8. Bibliografía	10

1. Introducción

1.1. Motivación

La generación precisa de imágenes territoriales con representaciones de altura detalladas es crucial para una variedad de aplicaciones, desde la planificación urbanística y la ingeniería civil hasta la gestión de recursos naturales y la respuesta a desastres. Sin embargo, los métodos tradicionales de generación de modelos de elevación del terreno (DEMs) suelen ser limitados en su capacidad para capturar la complejidad del relieve terrestre. En particular, las técnicas actuales carecen de la precisión necesaria para representar pendientes pronunciadas y otros elementos topográficos críticos. La motivación principal de este proyecto radica en la necesidad de superar estas limitaciones mediante la implementación de técnicas avanzadas de inteligencia artificial, específicamente el uso de redes generativas adversarias (GANs). Se necesita poder generar imágenes territoriales realistas con la representación de las alturas de cada punto en el mapa.

1.2. Problemática

Los algoritmos matemáticos encargados de realizar operaciones similares, debido a su naturaleza, no pueden generar pendientes muy pronunciadas, lo cual es esencial para una representación fiel del terreno. Esta limitación motiva la búsqueda de métodos alternativos capaces de superar esta barrera. Las redes generativas adversariales (GANs) han demostrado un potencial significativo en la generación de imágenes realistas, lo que las convierte en una opción prometedora para abordar este desafío.

1.3. Objetivos Generales

El objetivo principal de este proyecto es la creación de una GAN (Generative Adversarial Network) para la recreación de mapas con elevaciones del terreno a partir de imágenes satelitales.

1.4. Objetivos Específicos

1. Desarrollar una arquitectura de GAN adaptada para la generación de modelos de elevación del terreno:

- Diseñar y entrenar una pix2pix-GAN para la traducción de imágenes satelitales en mapas de elevación.
 - Evaluar la efectividad del modelo en la generación de pendientes pronunciadas y otros elementos topográficos complejos.
- 2. Recopilar y preparar un conjunto de datos de alta calidad para el entrenamiento del modelo:**
- Obtener imágenes satelitales y datos de elevación del terreno de fuentes confiables como Google Earth Engine.
 - Preprocesar los datos para asegurar su compatibilidad y adecuación para el entrenamiento de la GAN.
- 3. Validar y evaluar el desempeño del modelo generado:**
- Realizar pruebas extensivas del modelo en diversas áreas geográficas para asegurar su robustez y generalización.
 - Comparar los modelos de elevación generados con datos de referencia y evaluar su precisión.

2. Estado del Arte / Preliminares

Una revisión exhaustiva de la literatura sobre el uso de redes generativas adversariales para la traducción de imágenes y la generación de modelos de elevación del terreno es esencial para contextualizar este proyecto. Se analizarán trabajos previos y metodologías relacionadas para identificar las mejores prácticas y los desafíos más comunes.

En los últimos años, el uso de redes generativas adversarias (GANs) ha revolucionado el campo de la inteligencia artificial, particularmente en aplicaciones de generación y traducción de imágenes. Introducidas por Goodfellow et al. en 2014 [1], las GANs han demostrado ser eficaces en la creación de imágenes realistas a partir de datos de entrenamiento, superando las limitaciones de los métodos tradicionales.

2.1. Redes Generativas Adversarias (GANs)

Las GANs consisten en dos redes neuronales que compiten entre sí: el generador (G), que crea datos falsos a partir de un ruido aleatorio, y el discriminador (D), que intenta distinguir entre datos reales y generados. Esta configuración adversarial mejora continuamente ambos modelos, resultando en la generación de datos altamente realistas. Las aplicaciones de las GANs abarcan desde la generación de imágenes y videos hasta la síntesis de voz y la creación de obras de arte [2].

2.2. Pix2Pix GAN

Una de las variantes más destacadas de las GANs es la *pix2pix* GAN, presentada por Isola et al. en 2017 [3]. Esta arquitectura se utiliza específicamente para la traducción de imágenes de un dominio a otro, manteniendo la estructura y el contexto de la imagen de entrada. A diferencia de las GANs tradicionales, el modelo *pix2pix* emplea una red U-Net como generador y una red PatchGAN como discriminador. La U-Net permite la transferencia de información a diferentes escalas, mientras que la PatchGAN evalúa la autenticidad de pequeños parches de la imagen, en lugar de la imagen completa.

2.3. Aplicaciones de Pix2Pix GAN y CycleGan en Cartografía

El uso de *pix2pix* GANs en cartografía y geografía ha abierto nuevas posibilidades para la generación de modelos de elevación del terreno (DEMs) y la simulación de paisajes naturales. Trabajos como los de Isola et al. en 2017 [3] y Zhu et al. [4] han demostrado la capacidad de las GANs para traducir imágenes aéreas en mapas de calles y viceversa. Estos avances han permitido la creación de mapas más precisos y detallados, superando las limitaciones de los métodos tradicionales basados en interpolación y aproximaciones matemáticas.

2.4. Desafíos y Limitaciones

A pesar de sus éxitos, las GANs y, en particular, las *pix2pix* GANs, enfrentan varios desafíos. La necesidad de grandes cantidades de datos de entrenamiento de alta calidad, el riesgo de sobreajuste, y la inestabilidad durante el entrenamiento son problemas comunes. Además, la capacidad de las GANs para generar pendientes muy pronunciadas en modelos de elevación sigue siendo limitada. Sin embargo, los avances continuos en técnicas de entrenamiento y arquitectura de redes neuronales prometen mitigar estos desafíos en el futuro [5].

En resumen, la aplicación de *pix2pix* GANs en la generación de imágenes territoriales realistas representa un avance significativo en la cartografía moderna. La investigación actual se centra en mejorar la precisión y eficiencia de estos modelos, con el objetivo de crear representaciones del terreno cada vez más fieles a la realidad.

3. Propuesta de Solución

Se implementó una *pix2pix*-GAN que, dada una imagen satelital, genera una matriz de alturas del terreno. Los detalles técnicos de la implementación son los siguientes:

- **Input:** Imagen satelital de 3x256x256 (RGB) en formato GeoTIFF.
- **Output:** Mapa de alturas de 1x256x256.

Dada la imagen satelital y su combinación de colores (RGB), se calcula el tipo de terreno correspondiente. Cada archivo (imagen y altura) se transforma y se ajusta a dimensiones de 256x256, ya que ambos deben tener la misma dimensión debido a la naturaleza del algoritmo. Posteriormente, la matriz resultante del modelo se transforma a 32x32 para cumplir con las restricciones del problema.

3.1. ¿Qué es pix2pix-GAN?

Pix2Pix es un tipo de modelo de red generativa adversaria (GAN) diseñado para tareas de traducción de imágenes, es decir, convertir una imagen de un dominio a otro. Fue introducido en 2017 en el trabajo titulado "Image-to-Image Translation with Conditional Adversarial Networks". A diferencia de las GANs tradicionales, que generan imágenes a partir de un vector de ruido aleatorio, Pix2Pix toma una imagen de entrada y produce una imagen de salida correspondiente, lo que lo hace adecuado para aplicaciones como la conversión de bocetos en imágenes, la colorización de imágenes en blanco y negro, la conversión de fotos de día a noche, entre otros.

La arquitectura de Pix2Pix consta de dos componentes principales:

- **Generador (G):** Toma una imagen de entrada y genera una imagen que se espera se asemeje a una imagen realista del dominio objetivo. En Pix2Pix, el generador suele ser una red de tipo "U-Net", una arquitectura de red neuronal que permite la propagación de contextos a través de las diferentes escalas de la imagen.
- **Discriminador (D):** Su función es distinguir entre las imágenes reales del dominio objetivo y las imágenes generadas por el generador. El discriminador en Pix2Pix es una CNN (red neuronal convolucional) que clasifica segmentos de la imagen (en lugar de la imagen completa) como real o falsa, lo cual se conoce como PatchGAN.

El entrenamiento de Pix2Pix se realiza de manera adversaria: el generador intenta producir imágenes que el discriminador no pueda distinguir de las reales, mientras que el discriminador se mejora para distinguir entre imágenes reales y generadas. Este proceso se guía mediante una función de pérdida que combina la pérdida adversaria (para engañar al discriminador) y la pérdida L1 (para penalizar las diferencias pixel a pixel entre la imagen generada y la imagen real objetivo), ayudando así a producir resultados más fieles al dominio objetivo.

El resultado es un sistema capaz de transformar imágenes de un dominio a otro manteniendo la estructura y el contexto de la imagen de entrada, lo que lo hace muy efectivo para una amplia gama de aplicaciones de traducción de imágenes.

4. Base de Datos

La base de datos se obtuvo de Google Earth Engine:

- **Imágenes Satelitales:** LANDSAT/LC09/C02/T1.
- **Alturas:** CGIAR/SRTM90_V4.

La extracción de datos se realizó mediante la API de Google Earth Engine, con los datos almacenados en la cuenta de PixelCampione y descargados en la carpeta del *dataset* para su posterior uso. La región se dividió en partes debido a las restricciones de la API, requiriendo múltiples llamados a la API.

5. Resultados de las pruebas

El modelo fue probado con diferentes funciones de pérdidas tanto para el Discriminador como para el Generador. Debido a que la salida del modelo es una matriz de 256x256 de elevaciones, impide una evaluación visual del modelo, por tanto nos guiamos por la variación de las funciones de pérdida y la puntuación otorgada a las salidas generadas por el Discriminador.

5.1. Intentos de configuración

El problema principal de estos modelos es que durante las primeras iteraciones, el Discriminador era capaz de superar fácilmente al Generador, provocando que estuviera muy seguro de si la entrada era falsa o no. Dándole valores como 0.999 a las entradas reales y 0.001 a las falsas, restringiendo el aprendizaje del Generador.

5.1.1. L1 con BCEwithLogits

Epoch : 0 — Gen Loss : 7779.90283203125 — Disc Loss : 0.7112271785736084
 Epoch : 50 — Gen Loss : 7834.88427734375 — Disc Loss : 0.30028036236763
 Epoch : 100 — Gen Loss : 7570.71240234375 — Disc Loss : 0.06394217163324356

5.1.2. L1 with MSE

Epoch : 10 — Gen Loss : 8701.7890625 — Disc Loss : 0.33725571632385254
 Epoch : 20 — Gen Loss : 8048.34912109375 — Disc Loss : 0.30385181307792664
 Epoch : 30 — Gen Loss : 6809.49462890625 — Disc Loss : 0.2835003137588501
 Epoch : 40 — Gen Loss : 7663.60498046875 — Disc Loss : 0.2863680422306061
 Epoch : 50 — Gen Loss : 7830.93115234375 — Disc Loss : 0.22010774910449982
 Epoch : 60 — Gen Loss : 8029.484375 — Disc Loss : 0.2207767814397812
 Epoch : 70 — Gen Loss : 8668.478515625 — Disc Loss : 0.22636985778808594
 Epoch : 80 — Gen Loss : 8916.2041015625 — Disc Loss : 0.21985182166099548
 Epoch : 90 — Gen Loss : 7031.05810546875 — Disc Loss : 0.1898426115512848
 Epoch : 100 — Gen Loss : 7979.99951171875 — Disc Loss : 0.22319374978542328

5.1.3. MSE with BCEwithLogits

Epoch : 0 — Gen Loss : 798742.75 — Disc Loss : 0.7151925563812256
Epoch : 10 — Gen Loss : 99210.1484375 — Disc Loss : 0.7176052927970886
Epoch : 20 — Gen Loss : 82351.9453125 — Disc Loss : 0.5502198934555054
Epoch : 30 — Gen Loss : 107344.359375 — Disc Loss : 0.461466521024704

Al notar que el Discriminador tenía una curva de aprendizaje menos inclinada que en los modelos anteriores, se aplicó un entrenamiento en dos tiempos para lograr que tuviera la misma curva de aprendizaje que el Generador.

Epoch : 10 — Gen Loss : 5682.86572265625 — Disc Loss : 0.10323390364646912
Epoch : 20 — Gen Loss : 3714.49267578125 — Disc Loss : 0.10353855788707733
Epoch : 30 — Gen Loss : 3954.466796875 — Disc Loss : 0.11623755097389221
Epoch : 40 — Gen Loss : 5263.35107421875 — Disc Loss : 0.03681714087724686
Epoch : 50 — Gen Loss : 4207.08984375 — Disc Loss : 0.04253549873828888
Epoch : 60 — Gen Loss : 4234.42138671875 — Disc Loss : 0.06193991377949715
Epoch : 70 — Gen Loss : 5359.783203125 — Disc Loss : 0.01920810341835022
Epoch : 80 — Gen Loss : 3819.768798828125 — Disc Loss : 0.1254141926765442
Epoch : 90 — Gen Loss : 4847.0 — Disc Loss : 0.033486366271972656
Epoch : 100 — Gen Loss : 4832.95458984375 — Disc Loss : 0.027414310723543167
Epoch : 110 — Gen Loss : 4499.0068359375 — Disc Loss : 0.018689129501581192
Epoch : 120 — Gen Loss : 4022.912841796875 — Disc Loss : 0.03827538713812828
Epoch : 130 — Gen Loss : 4988.22509765625 — Disc Loss : 0.040250711143016815
Epoch : 140 — Gen Loss : 4469.6435546875 — Disc Loss : 0.06191401183605194
Epoch : 150 — Gen Loss : 4215.97412109375 — Disc Loss : 0.09380695223808289
Epoch : 160 — Gen Loss : 4244.1259765625 — Disc Loss : 0.014170262962579727
Epoch : 170 — Gen Loss : 5543.23486328125 — Disc Loss : 0.025407154113054276
Epoch : 180 — Gen Loss : 5570.6484375 — Disc Loss : 0.01545000821352005
Epoch : 190 — Gen Loss : 4534.845703125 — Disc Loss : 0.04580564424395561

Luego de aplicar el procedimiento se puede ver una marcada mejora de los valores de pérdida del Generador y del Discriminador

5.2. Modelo final

5.2.1. MSE con MSE con dos tiempos

Luego de varias pruebas se determinó que el modelo con los mejores resultados es el que tanto para el Generador como el Discriminador se les asignaron la función de pérdida del Error de Mínimos Cuadrados (MSE)

Para esta solución se obtuvieron resultados que eran un poco mejores que las demás. Se pudo observar como aumentaba la puntuación dada por el Discriminador a la matriz generada, y a su vez como este disminuía en las siguientes épocas. Pudiéndose observar el como tanto el Generador como el Discriminador se .enfrentaban”.

Al ser la mejor combinación encontrada hasta el momento, se utilizó un entrenamiento en dos tiempos. Este consiste en actualizar el generador más frecuentemente que el discriminador. Para este problema el Generador se entrenó dos veces por cada vez que lo hacía el Discriminador.

Esto se hizo con el fin de evitar el problema actual que era que el Discriminador superaba fácilmente al Generador.

5.2.2. Resultados

Epoch : 10 — Gen Loss : 5161.53466796875 — Disc Loss : 0.23925688862800598
Epoch : 20 — Gen Loss : 4585.37939453125 — Disc Loss : 0.12140040844678879
Epoch : 30 — Gen Loss : 5591.15771484375 — Disc Loss : 0.09408846497535706
Epoch : 40 — Gen Loss : 4471.14892578125 — Disc Loss : 0.09517452120780945
Epoch : 50 — Gen Loss : 4268.10205078125 — Disc Loss : 0.15849535167217255
Epoch : 60 — Gen Loss : 5619.7939453125 — Disc Loss : 0.053437717258930206
Epoch : 70 — Gen Loss : 4717.66552734375 — Disc Loss : 0.08003553748130798
Epoch : 80 — Gen Loss : 4510.0126953125 — Disc Loss : 0.0807599350810051
Epoch : 90 — Gen Loss : 4117.96728515625 — Disc Loss : 0.14058037102222443
Epoch : 100 — Gen Loss : 5128.8388671875 — Disc Loss : 0.06015884503722191
Epoch : 110 — Gen Loss : 4116.8330078125 — Disc Loss : 0.10317926108837128
Epoch : 120 — Gen Loss : 5011.46630859375 — Disc Loss : 0.039441488683223724
Epoch : 130 — Gen Loss : 3917.461181640625 — Disc Loss : 0.07097624242305756
Epoch : 140 — Gen Loss : 4299.22998046875 — Disc Loss : 0.06659030169248581
Epoch : 150 — Gen Loss : 3580.409912109375 — Disc Loss : 0.07947041839361191
Epoch : 160 — Gen Loss : 5353.79736328125 — Disc Loss : 0.054755087941884995
Epoch : 170 — Gen Loss : 4320.47265625 — Disc Loss : 0.08891814202070236
Epoch : 180 — Gen Loss : 4033.01318359375 — Disc Loss : 0.04331081360578537
Epoch : 190 — Gen Loss : 4119.53125 — Disc Loss : 0.051476653665304184
Epoch : 200 — Gen Loss : 4353.91064453125 — Disc Loss : 0.05317274481058121
Epoch : 210 — Gen Loss : 4696.810546875 — Disc Loss : 0.04456945136189461

Durante el entrenamiento se pudo observar como el Discriminador, a diferencia de los otros modelos, no sobrepasó al Generador. Se pudo observar cómo la puntuación dada al output generado variaba de 0.4 a 0.7 durante cada epoch. Sin embargo se sigue evidenciando cómo el Discriminador está siempre un poco por encima al Generador.

6. Discusión de los resultados

6.1. Repercusión ética de las soluciones

6.1.1. Privacidad y Seguridad

El uso de imágenes satelitales y datos geoespaciales puede plantear preocupaciones de privacidad, especialmente si se utilizan imágenes que incluyen áreas

habitadas o sensibles. Es esencial asegurarse de que los datos utilizados provienen de fuentes autorizadas y que se cumplen todas las normativas de privacidad y protección de datos vigentes. Además, los resultados generados por las GANs no deben comprometer la seguridad de individuos o comunidades, evitando la divulgación de información que podría ser utilizada con fines malintencionados.

6.1.2. Transparencia y Explicabilidad

Las técnicas basadas en IA, particularmente las GANs, son conocidas por su naturaleza de “caja negra”, donde los procesos internos pueden ser opacos y difíciles de interpretar. Esta falta de transparencia puede generar desconfianza entre los usuarios y limitar la aceptación de la tecnología. Es fundamental trabajar en la explicabilidad de los modelos, proporcionando información clara sobre cómo se generan los mapas de elevación y qué datos se utilizan. Los usuarios deben ser conscientes de las limitaciones y posibles errores del sistema para tomar decisiones informadas.

6.1.3. Equidad y Accesibilidad

La implementación de estas tecnologías debe asegurar que sus beneficios se distribuyen de manera equitativa y que no perpetúan o agravan desigualdades existentes. Por ejemplo, la disponibilidad y calidad de las imágenes satelitales pueden variar entre regiones, lo que podría resultar en una representación desigual del terreno y afectar negativamente a las comunidades menos favorecidas. Es crucial desarrollar políticas que garanticen un acceso equitativo a los recursos y tecnologías, así como mecanismos para corregir posibles sesgos en los datos.

6.1.4. Impacto Ambiental

El entrenamiento de modelos de IA, especialmente aquellos que requieren grandes cantidades de datos y poder computacional, puede tener un impacto ambiental significativo debido al consumo de energía. Es importante considerar la sostenibilidad y buscar formas de minimizar el impacto ambiental, como la optimización de los algoritmos y el uso de fuentes de energía renovable. Fomentar prácticas de desarrollo sostenibles es esencial para reducir la huella ecológica de estas tecnologías.

6.1.5. Uso Responsable de la Tecnología

Finalmente, es crucial promover un uso responsable de las GANs en la generación de DEMs. Los resultados de estos modelos deben ser utilizados con fines éticos y beneficiosos para la sociedad, como la planificación urbanística, la gestión de recursos naturales y la respuesta a desastres. Se debe evitar el uso de esta tecnología para actividades que puedan causar daño o explotación indebida de los recursos naturales o de las comunidades.

7. Conclusiones y trabajo futuro

Para mejorar y expandir este proyecto, se proponen las siguientes ideas:

1. **Entrenamiento de un modelo para clasificación de terrenos:** Se puede entrenar otra instancia del modelo donde la salida sea el tipo de terreno correspondiente. Esto se puede realizar de manera que a cada terreno le corresponda un número. Incluso se puede modificar el número asignado para que terrenos similares tengan representaciones numéricas cercanas. Luego se tendrían dos GAN, una para la generación de alturas y otra para la clasificación de terrenos.
2. **Generación de mapas de alturas sin imágenes satelitales:** Utilizando imágenes satelitales y su clasificación según el tipo de terreno, se podría generar un mapa de alturas sin necesidad de ingresar una imagen satelital. Esto implicaría la creación de vectores representativos de cada mapa en la base de datos, considerando la presencia de diferentes características del terreno. La vectorización de la consulta permitiría obtener mapas relacionados, facilitando la redacción e interpretación de consultas mediante técnicas de procesamiento de lenguaje natural o incluso usando ChatGPT.

8. Bibliografía

- **Pix2Pix:** <https://arxiv.org/abs/1611.07004>
- **GAN definition:** <https://arxiv.org/abs/1406.2661>
- **Google Earth catalog:** <https://developers.google.com/earth-engine/datasets/catalog>
- **Earth Engine API:** <https://developers.google.com/earth-engine/guides>
- **Dataset in personal Drive:** https://drive.google.com/drive/folders/1RnP7gD8rTcWRNzTr9LC65Z3RDC_HgxgX

Referencias

- [1] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. *Generative Adversarial Nets*. In Advances in Neural Information Processing Systems, 2014.
- [2] Antonia Creswell, Tom White, Vincent Dumoulin, Kai Arulkumaran, Biswa Sengupta, and Anil A. Bharath. *Generative Adversarial Networks: An Overview*. IEEE Signal Processing Magazine, 2018.

- [3] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros. *Image-to-Image Translation with Conditional Adversarial Networks*. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017.
- [4] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. *Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks*. In Proceedings of the IEEE International Conference on Computer Vision, 2017.
- [5] Karol Kurach, Mario Lucic, Xiaohua Zhai, Marcin Michalski, and Sylvain Gelly. *A Large-Scale Study on Regularization and Normalization in GANs*. In Proceedings of the 36th International Conference on Machine Learning, 2019.