Grenze International Journal of Engineering and Technology, June Issue

Survey on Indian Sign Language Video Generation

Satish Kamble[1], Srushti Mahajan[2], Komal Pathare[3], Aniket Sonkamble[4] and Shubham Shinde[5]
[1-5]Department of Information Technology, Pune Vidyarthi Griha's College of Engineering and Technology & G.K. Pate (Wani) Institute of Management, Pune, India
Email: satkam53@gmail.com, srushtimahajan2943@gmail.com , patharekomal39@gmail.com , aniketsnkamble07@gmail.com , shindeshubham1313@gmail.com

***Abstract*—Indian Sign Language serves as a primary mode of communication for the hearing-impaired community in India. Over the years, extensive research has been conducted to automate Indian Sign Language gesture generation using various computational approaches. This literature survey provides a comprehensive analysis of existing methods, frameworks, and technologies employed in ISL gesture generation. It reviews state-of-the-art techniques spanning speech-to-text conversion, natural language processing strategies, and gesture synthesis using animation tools such as Hamburg Sign Language Notation System and Signing Gesture Markup Language. The paper also examines deep learning models, including convolutional and recurrent neural networks, for gesture recognition and generation. A detailed comparison of methodologies, datasets, and performance evaluation metrics is presented to identify current trends and limitations in the field. Additionally, the survey discusses key challenges such as data scarcity, linguistic variations, and real-time processing constraints. The study aims to provide insights into the progress made in ISL gesture generation and outlines future directions for developing more robust and user-friendly assistive technologies.**

***Index Terms*— Indian Sign Language (ISL), Literature Survey, Gesture Generation, Natural Language Processing (NLP), HamNoSys, SiGML, Assistive Technology.**

I. INTRODUCTION

Education is a fundamental right, a key element of human development, and an essential tool for personal and societal growth. Still, despite its significance, millions of people worldwide continue to encounter major obstacles when trying to obtain high-quality education. Among the most affected are the deaf and mute communities, who often struggle to participate in mainstream education systems due to a lack of adequate resources, inclusive teaching methods, and communication support. In addition to limiting their access to education, the lack of a platform that precisely meets their needs also hinders their chances for social inclusion, professional progress, and personal development.

The difficulties faced by the deaf and mute communities are especially severe in India, where more than 63 million individuals have severe hearing loss. About 466 million individuals worldwide, 34 million of them are children, are estimated by the World Health Organization to have a debilitating hearing loss. These staggering numbers highlight the urgency for education systems to evolve and accommodate individuals with disabilities. Unfortunately, existing educational infrastructures are not equipped to address the unique needs of the deaf and mute, often leaving them behind in an increasingly knowledge-driven world.

Given the increasing importance of digital learning platforms, there is an urgent need for a system that

Grenze ID: 01.GIJET.11.2.563

accommodates the deaf and mute community. This survey aims to find such ideas, utilizing advanced technologies such as Natural Language Processing (NLP), gesture recognition, and animation generation to ensure that deaf and mute students can access the same quality of education as their hearing peers.

A. Sign Language

Across the world, there are many different sign languages. Just as each country has its own spoken language, people with hearing disabilities often prefer using their local sign language. As in India, we have Indian Sign Language (ISL), in Britain, British Sign Language (BSL), in USA American Sign Language (ASL), etc. Regional variations exist within sign languages, reflecting local expressions and customs. Although there is an international sign language intended as a global standard, it is not widely adopted.

II. RELATED WORK

When people without hearing disabilities try to communicate with the deaf community, gesture recognition technology is often used. Many researchers have worked on developing solutions in this area, primarily to help hearing individuals communicate with the deaf. However, deaf individuals still face challenges in fully understanding the information being conveyed by sound.

III. LITERATURE SURVEY

A. Real-Time Translation of Arabic Video Content to Arabic Sign Language[1]

The paper introduces *Al Banan*, a PC-based application that translates Arabic speech into ArSL in real-time using a 3D avatar, aiding the deaf and hearing-impaired in understanding online media, including educational videos and live classes.

Technologies Used: Developed using Python and C#

- Speech Recognition: Converts video audio to text using Microsoft Cognitive Speech Services.
- Text Preprocessing: Tokenizes text, removes punctuation, and identifies compound words for accurate sign translation.
- Morphological Analysis: Uses CAMeL to extract linguistic features like lemma, part of speech, and number for better sentence structuring.
- ArSL Structure: Restructures text per ArSL grammar rules from linguistic studies.
- Dictionary Search: Matches processed text to ArSL dictionary files; uses finger spelling for unmatched words.

Arabic Sign Language and Its Translation

- ArSL Dictionaries: Two primary ArSL dictionaries in Saudi Arabia—one by the Princess Al-Anoud Charitable Foundation and another by the Saudi Society for Hearing Impairment—contain sign descriptions.
- Sign Language Structure: ArSL grammar, syntax, and word order differ from spoken Arabic and vary across cultural and linguistic contexts.

Results

The prototype translated a 45-second Arabic video into ArSL with 66% accuracy. Among eight deaf testers, only two fully understood the content due to limited writing skills. Avatar speed was effective for 75% of users but lacked speed control, requiring further improvements in accuracy and user customization.

B. Indian Sign Language Generation – A Multi-modal Approach[2]

This paper presents a method to generate ISL gestures from text, audio, and video, improving accessibility for the hearing impaired. It automates text translation into ISL gestures, displayed through animations for better understanding.

Technologies Used

- Natural Language Processing (NLP): Uses the Stanford Parser for text parsing, POS tagging, and lemmatization to break sentences into ISL-compatible units.

Deep Learning Models

- CNNs extract spatial features from video frames for ISL gesture recognition.
- RNNs analyze temporal features to predict ISL gestures.
- SiGML Files: Maps words to ISL gestures for accurate animation generation.

- Google Speech-to-Text API: Converts spoken audio into text for ISL translation.
- Pytesseract OCR API: Extracts text from images for ISL conversion

Animation Tools

Text input is processed using NLP techniques, applying ISL grammar rules like stop-word removal, tokenization, and lemmatization. The processed text maps to SiGML files, which generate ISL animations. Various input types are supported:

- Text: Directly processed.
- Video: Subtitles extracted via YouTube Data API.
- Audio: Converted to text using Google's Speech-to-Text API.
- Image: Text extracted via OCR.

Parsing identifies grammatical structures, adjusting sentence formats for ISL. Stop words are removed, and lemmatization ensures word compatibility. If a word lacks a SiGML file, character-based mapping is used for animation.

For ISL Recognition, video input is analyzed by CNNs and RNNs to extract spatial and temporal features, ensuring accurate sign recognition.

Results

The ISL Generation system allows users to input text, video, audio, or images. Text input is converted into ISL animations, while video generates ISL alongside playback. Audio input is recorded and processed into animations, and text in images is converted similarly. The output is displayed via an animated avatar.

In ISL Recognition, sign gesture videos are analyzed by converting them into sequential frames. Model performance is evaluated using a confusion matrix and accuracy tracking over epochs. The proposed CNN+LSTM RNN model achieved 96% accuracy, outperforming Fuzzy C-Means with HMM (75%) and 3DCNN (72.32%).

C. A Web-Based Audio-to-Sign Language Converter for Chinese National Sign Language [3]

The web-based Chinese National Sign Language (CNSL) converter provides an online platform to translate spoken Chinese into sign language using speech recognition, text processing, and animation rendering. The system captures audio, converts it to text, and maps the result to CNSL gestures, displayed via a 3D avatar.

Workflow

- System Development: A web-based application using HTML, CSS, and JavaScript for the frontend and Python with Flask or Django for the backend, allowing users to select input types and view results easily.
- Audio Collection & Text Conversion: Captures audio from the user's microphone and converts it into text using the Google Speech-to-Text API.
- Text Processing & Segmentation: Uses HanLP for Chinese word segmentation, breaking text into meaningful units.
- Database Mapping & Video Retrieval: The system maps segmented words to sign language video paths stored in a MySQL or PostgreSQL database, retrieving them using SQL queries.
- Video Generation & Display: The videos are displayed using HTML5 Video with JavaScript for user interactions like replay.

Results

After three rounds of testing (each with 10 trials), the system achieved a 90% success rate (9 out of 10 tests successful). Factors affecting accuracy included background noise, speech recognition efficiency, segmentation effectiveness, and database limitations when encountering unknown phrases.

D. Bilingual Social Robot with Sign Language and Natural Language [4]

As technology advances, robots are increasingly becoming a part of our daily lives, serving in fields such as education, therapy, and entertainment. In human-robot interaction (HRI), speech is the most natural form of communication, facilitating clear exchanges and enhancing engagement through tone and emotion. However, for individuals with hearing impairments, speech alone may not be effective. Sensory compensation allows deaf individuals to develop heightened proficiency in visual and tactile modalities, making sign language their primary means of communication. Thus, the need for a bilingual robot that can communicate using both natural speech and sign language is crucial for bridging the communication gap and fostering inclusivity in diverse social, educational, and professional settings. This paper explores a framework for such a robot, utilizing

multimodal communication to cater to both deaf and non-deaf individuals simultaneously.

Technologies Used
The framework for the bilingual social robot incorporates various technologies to generate speech, gestures, and sign language animations.
The key technologies include:
1. Microsoft Azure TTS (Text-to-Speech):Converts text input into natural speech output.
2. Gesture Generation Model:Consists of a generator (temporal encoder and decoder) and a discriminator to produce gestures aligned with speech audio.
3. Text2Sign Tool:Generates 2D sign language poses from text input.
4. MocapNet:Estimates 3D human pose from 2D joint positions, producing BVH format for 3D sign language representation.
5. Mixamo Database:Provides animated 3D characters for visualizing sign language gestures.
6. Pepper Robot:Displays generated speech, gestures, and sign language animations via an integrated tablet.

Methodology
The framework's methodology is structured in three key stages:
1. Natural Language and Gesture Generation – Text input is processed through Microsoft Azure TTS to generate speech. This speech audio is then fed into a gesture generation model, which consists of a temporal encoder and decoder. Additionally, a discriminator is employed to assess the alignment between the generated gestures and speech, refining the accuracy of the output.
2. Sign Language Generation – Text input is processed through the Text2Sign tool to obtain 2D sign language poses. These 2D poses are further converted into 3D representations in BVH format using MocapNet. The generated 3D poses are then applied to a human mesh obtained from the Mixamo database, and the final avatar is displayed on the Pepper robot's tablet.

Natural Language and Sign Language Alignment – Speech, gesture, and sign language animations are generated separately. Future efforts will focus on developing synchronization techniques to ensure seamless multimodal interaction.

Result
The proposed bilingual social robot framework demonstrates the potential to revolutionize human-robot interaction by offering both natural language and sign language communication. Preliminary findings suggest feasibility, but further research is required to optimize synchronization and evaluate the system's effectiveness through user studies. Future work will focus on refining alignment techniques and integrating Large Language Models (LLMs) to enhance the robot's responsiveness to complex inquiries, ultimately advancing accessibility and inclusivity in diverse settings.

E. Indian Sign Language Generation from Live Audio or Text for Tamil[5]

In order to improve communication between hearing and hearing-impaired people, this research suggests a method that converts Tamil speech into Indian Sign Language (ISL).The system leverages speech recognition, translation, and gesture mapping to enable seamless communication in real-time. The integration of deep learning techniques such as Long Short-Term Memory (LSTM) and Bidirectional LSTM (Bi-LSTM) models for speech recognition and ISL gesture generation presents a novel approach to addressing the challenges of bridging communication gaps in diverse social settings.

Technologies Used
The proposed system utilizes the following technologies to convert Tamil speech into ISL:
- LSTM and Bi-LSTM Models: Deep learning models used for speech recognition, converting Tamil audio to text.
- Google API: Used for speech-to-text conversion, achieving high accuracy in speech recognition.
- Librosa Library: Used for audio pre-processing, feature extraction, and Mel Frequency Cepstral Coefficients (MFCC) extraction.
- Natural Language Processing (NLP): Used for text processing, including tokenization, stemming, and lemmatization.
- ISL Gesture Animation Dataset: A custom dataset consisting of animated gestures for ISL, used for mapping Tamil text to corresponding sign language gestures.
- Python Programming: The entire system is implemented using Python, leveraging libraries like Tens or Flow,

Keras, and Google Cloud APIs.

Methodology

1. Audio Pre-processing and Speech Recognition – The user provides Tamil speech input, which is captured through a microphone. The audio is pre-processed using the Librosa library to extract features such as MFCCs, which are then fed into an LSTM or Bi-LSTM model to convert speech to Tamil text. Additionally, the Google API is employed for comparison to ensure accuracy.
2. Text Translation and Sign Language Gesture Mapping – The generated Tamil text is translated into English using the Google API. The English text is then processed using NLP techniques to extract meaningful words, which are mapped to ISL gestures using a gesture dataset. If a direct match is found, the corresponding animated ISL gesture is displayed; otherwise, the system generates a finger-spelled representation.
3. Output Generation – The system displays the translated ISL gesture video or finger-spelled letters, allowing the user to view the corresponding sign language animation.

Result

The speech-to-sign language system was evaluated using a video GIF dataset and compared deep learning models (LSTM, Bi-LSTM) with the Google API. The Google API outperformed the deep learning models, achieving 95% accuracy, while the models showed limited performance due to the small dataset size. The system categorizes outputs into manual and non-manual signs or finger-spelled signs. It performs well with common phrases but struggles with rare words due to dataset limitations. The experiment was conducted on a system with an Intel i7 processor and 8 GB RAM.

F. Real-Time Speech To Sign Language Translation Using Machine And Deep Learning

This paper proposes a system for real-time speech-to-sign language translation using machine and deep learning.

Technologies Used

- Deep Learning Models: CNNs, RNNs, and LSTM networks are employed for speech recognition and gesture generation.
- Speech-to-Text Conversion: Speech is converted into text using advanced models like Google API for speech recognition.
- Feature Extraction: Mel-Frequency Cepstral Coefficients (MFCC), Fast Fourier Transform (FFT), and Discrete Cosine Transform (DCT) are used for audio feature extraction.
- Natural Language Processing (NLP): NLP techniques such as tokenization and lemmatization are used for processing the text and generating meaningful phrases.
- Sign Language Gesture Dataset: A custom dataset of animated gestures representing sign language is utilized for mapping text to gestures.
- Python Programming: Python is used for implementing the entire system, leveraging libraries like TensorFlow, Keras, and other speech processing APIs.

Methodology

1. Audio Pre-processing and Speech Recognition, 2. Text Translation and Gesture Mapping and 3. Output Generation.

Results

The real-time sign language translation system demonstrated an accuracy of 93.4%, successfully integrating components like voice recognition, text-to-sign translation, and animation through a micro service architecture. The system efficiently handled common speech inputs, translating them into Indian Sign Language (ISL) gestures with minimal latency. While it struggled with rare or uncommon words due to limited dataset size, the micro services architecture ensured scalability, fault isolation, and easy maintenance, making the system stable and responsive. Overall, it shows great potential in providing real-time accessibility for the hearing-impaired and can be further refined for improved performance and inclusivity.

G. Sign Language Generation System Based on Indian Sign Language Grammar [8]

This paper presents a novel system that uses cutting-edge linguistic and animation technologies to translate English text into Indian Sign Language (ISL).

Technologies Used

- Natural Language Processing: To guarantee proper structure, a specialised ISL parser analyses English text, finds important terms, and applies ISL grammatical rules.

- Using standardised symbols, the Hamburg Notation System (HamNoSys) offers a method for transcribing sign language motions.
- For simple animation, the Signing Gesture Mark-up Language (SiGML) transforms HamNoSys symbols into XML format.
- Real-time 3D avatar animation helps users better comprehend the information by bringing sign language to life.

Methodology
ISL Parser: English text is entered by the user.The algorithm follows grammar rules and eliminates extraneous components like punctuation.Words are reordered to match ISL sentence structure.
HamNoSys Notation: Hand shapes, orientations, and movements are represented using standardised symbols.For instance, certain HamNoSys symbols are used to represent the sign for "alone".
SiGML Conversion: To facilitate animation, the HamNoSys representation is transformed into SiGML.
Avatar Animation:Users can view the translated sign language in real time through a digital avatar.

Results
The proposed system was evaluated using the BLEU score, achieving an accuracy of 0.95 by comparing system-generated ISL translations with expert-generated ones. User evaluations indicated a word accuracy rating of 4.2 out of 5 and a sentence accuracy rating of 3.8 out of 5, with users providing positive feedback on the system's usability. The system offers an innovative solution for translating English text into ISL using a 3D avatar, enabling real-time communication support. However, certain limitations were identified, including the absence of facial expressions, challenges in processing complex sentences, and a limited vocabulary. Future enhancements will focus on improving sentence translation accuracy, incorporating facial animations, and expanding the database to ensure better language coverage and user experience.

H. Multi Facet: A Multi-Tasking Framework for Speech-to-Sign Language Generation[9]

Multi Facet aims to bridge this gap by translating speech into sign language using advanced technology. By combining speech rhythm (prosody), text semantics, and facial expressions, it creates a more natural and comprehensive representation of sign language, making communication more inclusive.

Technologies Used
- Speech Recognition: Tacotron 2 GST captures the rhythm, tone, and emotional nuances of speech.
- Text Understanding: BERT embeddings extract meaning from text to ensure accurate translation.
- Facial Action Units (FAUs): These detect subtle facial muscle movements, essential for expressing emotions and enhancing communication in sign language.
- Sign Pose Generation: A transformer-based model generates hand, face, and upper body movements, translating speech into visual language.

Indian Sign Language and Its Translation:
Indian Sign Language relies heavily on non-verbal elements, such as facial expressions, body language, and precise hand gestures. MultiFacet incorporates these elements using a keypoint-based system that maps movements to sign language. However, challenges remain, particularly in capturing intricate details like finger movements and subtle facial expressions, which are critical for meaning.

Results
The prototype showed promising results, outperforming older methods like Speech2Sign. A Dynamic Time Warping (DTW) score of 13.23, indicating better alignment between generated poses and actual sign language. A Probability of Correct Keypoints (PCK) score of 81%, reflecting accurate placement of facial and body keypoints. While the system effectively captures broad gestures, it struggles with fine-grained details. Testing with sign language users also highlighted gaps, such as occasional inaccuracies in hand and finger positions and the need for smoother transitions between movements.

I. GAN Based Indian Sign Language Synthesis[11]

Sign Language Synthesis (SLS) is a complex and essential task for improving accessibility for the hearing impaired population, specifically in languages like Indian Sign Language (ISL). The challenge of generating realistic and controllable visual representations of sign language has long been addressed using computer animation. However, these traditional methods come with significant limitations, including the need for expensive motion capture technology and extensive manual work to clean noisy data. With the advent of

Generative Adversarial Networks (GANs), new methods for pose and motion transfer have emerged, offering promising results in creating more efficient and scalable solutions for SLS.
In this work, the authors propose a GAN-based SLS model tailored for ISL, focusing on generating high-quality, continuous sign language sequences without the need for large annotated datasets. This model addresses challenges such as the lack of good-quality hand images and the creation of smooth, continuous signing sequences from individual signs in a video lexicon. Their approach represents a significant advancement in the field, leveraging GANs to perform continuous sign language synthesis without relying on manually annotated data.

Technologies Used

- Generative Adversarial Networks (GANs): The model uses GANs for conditional image generation and motion transfer. GANs consist of a generator and a discriminator, which work together to produce realistic outputs, such as images or videos, from given inputs. This approach is utilized for generating hand and body poses in sign language synthesis.
- Pose Estimation Models (OpenPose, MediaPipe): MediaPipe is especially used for hand pose estimate, whereas OpenPose is utilized for full-body pose estimation. These models are crucial for extracting skeleton data from videos, which is then used to generate the corresponding sign language output.
- Affine Transformation: This technique is used to normalize the pose data to accommodate variations in the physical appearance and body shape of different signers. It ensures that the generated sign language is consistent, regardless of the individual performing the sign.
- Smoothing Networks: To address the issue of artifacts arising from combining outputs from separate GAN generators (one for hands and one for the body), the authors employ a smoothing network. This network helps blend the generated body and hand images into a cohesive and realistic output.
- Sign Stitching (Rule-based approach): This method is used to generate continuous sign language output by combining individual signs from a lexicon, using video segments without the need for large-scale annotated data.

Methodology

Pose Extraction and Normalization

The model begins by extracting skeleton pose data from videos using pre-trained models like OpenPose (for full-body poses) and MediaPipe (for hand poses). This data is essential for generating accurate sign language output.
The pose data is then normalized using affine transformations to account for variations in body shape and pose. This ensures the output video has consistent body positioning, independent of the performer.

Hand and Body Generation

Two separate GAN generators are used: one for generating the body pose and the other for generating the hand pose. This separation is necessary to maintain clarity and accuracy, as handshapes are crucial for intelligibility in sign language.
The outputs from both generators are combined. However, since combining outputs from different GANs can lead to artifacts, a smoothing network is employed to eliminate any inconsistencies, ensuring a seamless video output.

Continuous Sign Language Output

To produce a continuous sign language video, the authors use a method called "sign-stitching," where individual signs are stitched together from a publicly available ISL video lexicon.
A rule-based approach is employed to detect the onset and end of each sign, and interpolation techniques are applied to ensure smooth transitions between signs, producing a fluent and natural signing sequence.

Evaluation and Testing

The proposed GAN-based model is evaluated both quantitatively and qualitatively, through human perception tests, to assess the realism, intelligibility, and fluidity of the generated sign language sequences. The model is found to perform competitively with other state-of-the-art SLS methods.
This approach represents a novel and resource-efficient method for generating Indian Sign Language sequences, addressing the challenges of data scarcity and manual annotation in a practical and scalable manner.

IV. CONCLUSIONS

In order to close the communication gap between the hearing and deaf communities, the survey on sign language creation highlights developments in methods such as machine learning and 3D pose identification. While progress has been made in isolated tasks, challenges persist in real-time translation, context comprehension, and

natural gesture production. Future work should focus on refining models, improving cross-lingual translation, and developing practical, user-friendly applications to make sign language technology more accessible and impactful.

REFERENCES

[1] A. Maghraby, S. Qutub, A. Alandijani, H. Alandijani, L. Bakhsh, and N. Alharbi, "A Real-Time Translation of Arabic Video Contents to Arabic Sign Language," 2021 International Conference on Computational Science and Computational Intelligence (CSCI), pp. 1008–1013, Dec. 2021, doi: https://doi.org/10.1109/csci54926.2021.00082.

[2] P. Chaudhari and Mangesh Bedekar, "Indian Sign Language Generation – A multi-modal approach," 2022 13th International Conference on Computing Communication and Networking Technologies (ICCCNT), vol. 12, pp. 1–6, Jul. 2023, doi: https://doi.org/10.1109/icccnt56998.2023.10306461.

[3] P. Wang, N. Yin, and Krishamoorthy Sujatha, "A Web-Based Audio-to-Sign Language Converter for Chinese National Sign Language," pp. 1045–1048, Dec. 2022, doi: https://doi.org/10.1109/iaecst57965.2022.10061950.

[4] X. Hei, C. Yu, H. Zhang, and A. Tapus, "A Bilingual Social Robot with Sign Language and Natural Language,", HRI '24 Companion, pp. 1–4, March 2024

[5] B. R. Reddy, D. C. Rup, Mathi Rohith, and Meena Belwal, "Indian Sign Language Generation from Live Audio or Text for Tamil," 2022 8th International Conference on Advanced Computing and Communication Systems (ICACCS), pp. 1507–1513, Mar. 2023, doi: https://doi.org/10.1109/icaccs57279.2023.10112880.

[6] D. Rai, N. Rana, N. Kotak, and M. Sharma, "Real-Time Speech to Sign Language Translation Using Machine and Deep Learning," 2022 10th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO), pp. 1–5, Mar. 2024, doi: https://doi.org/10.1109/icrito61523.2024.10522437.

[7] SUGANDHI, PARTEEK KUMAR, and SANMEET KAUR, "Sign Language Generation System Based on Indian Sign Language Grammar ,"ACMTrans. Asian Low-Resour. Lang. Inf. Process.,1-26,April 2020

[8] M.Kanakanti, S. Singh, and M. Shrivastava , "MultiFacet: A Multi-Tasking Framework for Speech-to-Sign Language Generation," ICMI '23 Companion, pp. 1–9, Oct. 2023.

[9] J. Huang, J. Chaijaruwanich, and V. Chouvatut, "Video-based Sign Language Recognition with R(2+1)D and LSTM Networks," 2024 16th International Conference on Knowledge and Smart Technology (KST), pp. 214–219, Feb. 2024, doi: https://doi.org/10.1109/kst61284.2024.10499646.

[10] S. Krishna, J. Ukey, and D. Babu, "GAN Based Indian Sign Language Synthesis," ICVGIP'21, pp. 1–8, Oct. 2021.