Game Voice Control

stendhal.syndrome.studio@gmail.com

Contents

1. About

Game Voice Control Plugin is offline recognition of speech commands in different languages.

Features:

- Does not require Internet connection;
- Supports **9** different languages;
- Recognition of made-up words (like “FUS RO DAH”);
- Activation of speech recognition by means of key word (similar to «Ok Google!» or «Siri»);
- Level adjustment of voice activity detection;
- Multiplatformity;
- Easiness of integration.

Supported languages:

- English [Dictionary];
- Russian [Dictionary];
- German [Dictionary];
- French [Dictionary];
- Spanish [Dictionary];
- Italian [Dictionary];
- Dutch [Dictionary];
- Hindi [Dictionary];
- Portuguese [Dictionary].

Supported platforms:

- Windows 10, Windows 7 Service Pack 1;
- Android (armeabi-v7a or arm64-v8a).

To enable the plugin on Windows 10 (Windows 7 Service Pack 1) it is required to install **Microsoft Visual C++ Redistributable Package 2015** (https://www.microsoft.com/ru-RU/download/details.aspx?id=48145)

Links:

Video instruction on plug-in usage.

Demo. Video

Demo.Video.Made-up words recognition

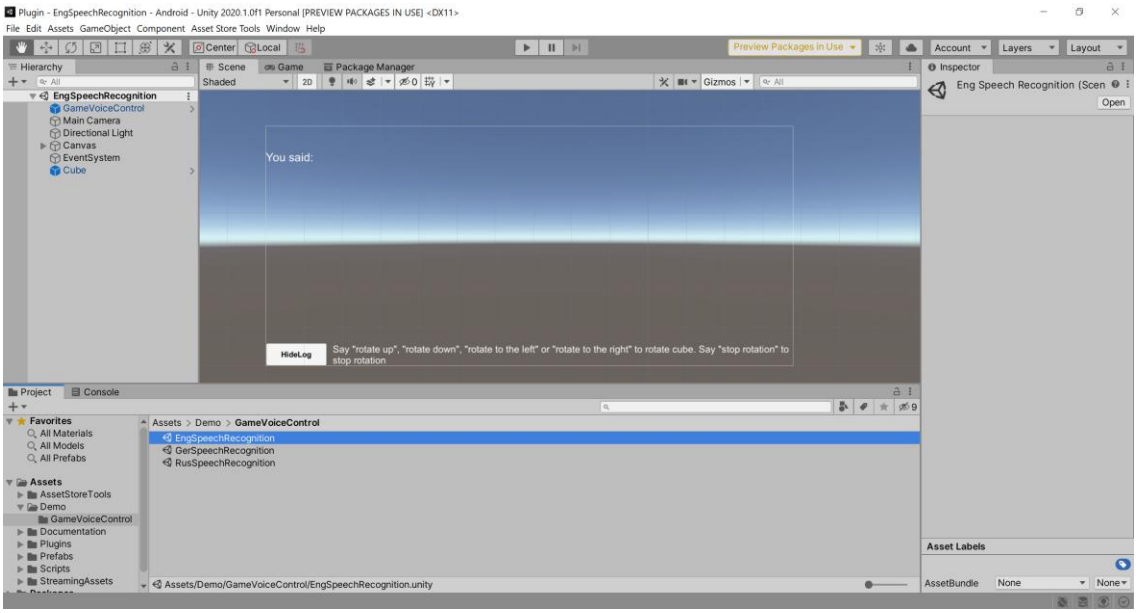
2. Quick start

2.1 Import package **GameVoiceControl.unitypackage**.

2.2 Make sure that the microphone is switched on and in working order;

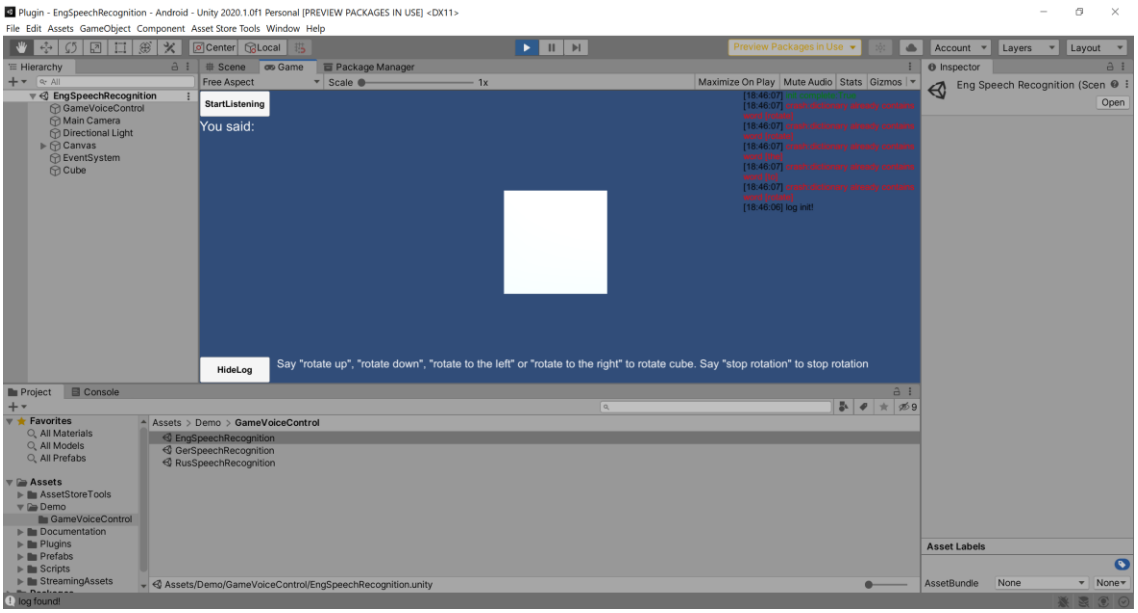
2.3 Enter settings of the microphone and adjust them: sensitivity is to be at zero;

2.4 Load the scene **EngSpeechRecognition** (GerSpeechRecognition, RusSpeechRecognition) from folder **Scenes**;



2.5 Press button  to load the scene;

2.6 In case of successful initialization there will appear button «**StartListening**» in the top left corner. The log is depicted in the right part of the scene. The list of commands is specified at the bottom of the screen; the result of recognition is depicted in the top left corner of the screen («**You said: …**»)

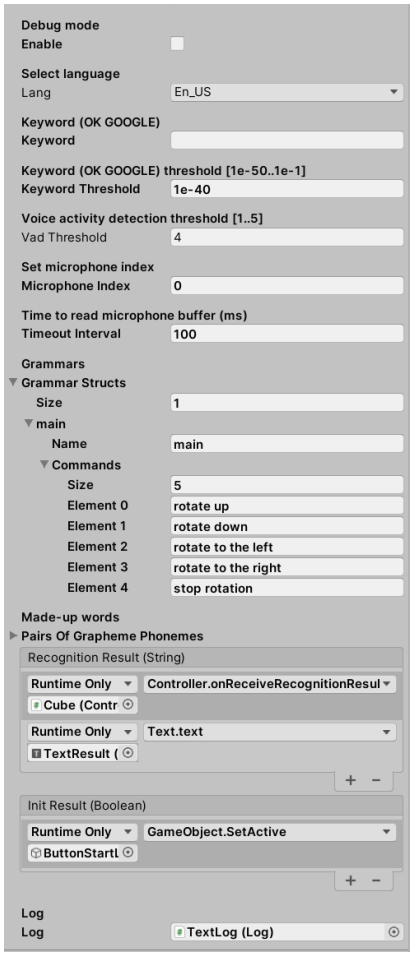


2.7 Press button «**StartListening**» to start speech recognition;

2.8 Say the command (e.g. «Rotate to the left») into the microphone. In case of successful recognition the cube will start to rotate in the corresponding direction. Say other commands – control the result of recognition.

3. API overview

Locate prefab **SpeechRecognition** from folder **Prefabs** on the scene and choose object **SpeechRecognition**.



Fields of class GameVoiceControl (Script):

- **Debug mode Enable** (*bool*) – enables the mode of log record to the file;
- **Lang** (*enum*) – to choose language model;
- **Keyword** (*string*) – key word (similar to «OK Google» or «Siri») to activate recognition system. After detection of the key word the system switches to recognition of the first set of commands from array **GrammarStructs**. After recognition of the command from the array **GrammarStructs** the system switches to the search of the key word. The parameter is not obligatory;
- **Keyword threshold** *(float)* - threshold of reaction of recognition system to a certain keyword. Range is from **1e-50** to **1e-1**. The smaller the value is the more sensitive the recognition system is and the larger the quantity of false-positive reactions is;
- **Vad Threshold** (*double*) – threshold of reaction of recognition system to the human speech. Range is from 1.0 to 5.0. The smaller the value is the more sensitive the recognition system is and the larger the quantity of false-positive reactions is. It is not recommended to change the value of the parameter;
- **Microphone Index** – the number (index) of the microphone that will be used during the work of the plugin.
- **TimeoutInterval** (*double*) – interval of reading microphone buffer. It is not recommended to change the value of the parameter;
- **Grammars** – array of set of commands. Parameter **Size** (*int*) for **Grammar Structs** specifies the quantity of sets of commands. Parameter **Name** (*string*) specifies the name (**key**) for the set of

commands. Parameter **Commands** – set of commands for recognition. Parameter **Size** (*int*) for **Commands** specifies the quantity of commands for recognition. Parameters **Element0 – ElementN** (*string*) – commands for recognition;

- **Made-up words** – array of pairs of grapheme phonemes. Parameter **Size** (*int*) for **Pairs Of Grapheme Phonemes** specifies the quantity of sets of made-up words. Parameter **Grapheme –** the word for recognition. Parameter **Phoneme –** transcription (phoneme) of the word that needs to be recognized.

  Example of PairOfGraphemePhoneme:

  Grapheme – «**bandersnatch**», Phoneme – «**B AE N D ER S N AE CH**»

- **Log** – **GameObject** with components **UnityEngine.UI.Text** and **Log** (/Assets/Scripts/Test/Log.cs). The parameter is not obligatory.

Methods of class GameVoiceControl (Script):

- **onStartListening()** – start of recognition of speech commands;
- **onStopListening()** – end of recognition;
- **getResult(string)** – processing of results of recognition;
- **initComplete(bool)** – processing of results of initialization of the recognition system;
- **getLogMessages(string)** – processing of messages from the recognition system;
- **getCrashMessages(string)** – processing of messages about errors of the recognition system;

Class events of GameVoiceControl (Script):

- **RecognitionResult (String)** – event (**UnityEvent<String>**) with results of speech recognition;
- **InitResult (Boolean)** – event (**UnityEvent<bool>**) with results of initialization of the speech recognition system.

For a more «delicate» adjustment of the recognition system use the private field `_speechRecognizer (class SpeechRecognizer of namespace UnityEngine.SpeechRecognition).`

4. Adding of made-up words

To get transcription (phoneme) of the word for the **English acoustic model** you may use the service LOGIOS Lexicon Tool

To get transcription (phoneme) of the word for the **Russian acoustic model** you may use the service ru4sphinx

To create transcriptions of made-up words **manually** you may use transcriptions from phonetic dictionaries as examples.

5. Future plans

- Adding of new languages (Mandarin, Kazakh, Greek)
- Adding of new platforms (Linux, iOS, Mac)

6. Troubleshooting

The testing was done on **Unity 2018.3 – 2020**. Work of plugin on earlier version is possible, but not guaranteed.

If the quantity of wrong results returned by the system is too large, try to decrease microphone sensitivity or increase **voice activity detection** in **GameVoiceControl (Script)**.

There is a peculiarity about commands in Hindi that are to be recognized. The words in the Hindi dictionary are given not Hindi letters but English ones by means of transliteration.

7. Contact us

Do you meet issues while using this plugin?

Do you have suggestions how to improve API?

Feel free to contact us: **stendhal.syndrome.studio@gmail.com**