

AI

SUNAPI

v2.6.2

2023-04-07



Copyright

© 2023 Hanwha Vision Co., Ltd. All rights reserved.

Restriction

Do not copy, distribute, or reproduce any part of this document without written approval from Hanwha Vision Co., Ltd.

Disclaimer

Hanwha Vision Co., Ltd. has made every effort to ensure the completeness and accuracy of this document, but makes no guarantee as to the information contained herein. All responsibility for proper and safe use of the information in this document lies with users. Hanwha Vision Co., Ltd. may revise or update this document without prior notice.

Contact Information

Hanwha Vision Co., Ltd.

Hanwha Vision 6, Pangyo-ro 319beon-gil, Bundang-gu, Seongnam-si, Gyeonggi-do, 13488, KOREA
www.hanwhavision.com

Hanwha Vision America

500 Frank W. Burr Blvd. Suite 43 Teaneck, NJ 07666
hanwhavisionamerica.com

Hanwha Vision Europe

Heriot House, Heriot Road, Chertsey, Surrey, KT16 9DT, United Kingdom
hanwhavision.eu

Hanwha Vision Middle East FZE

Jafza View 18, Office 2001-2003, Po Box 263572, Jebel Ali Free Zone, Dubai, United Arab Emirates
www.hanwhavision.com/ar

Table of Contents

| | |
|---|----|
| 1. Overview | 4 |
| 1.1. Description | 4 |
| 2. Metaattributearch | 5 |
| 2.1. Description | 5 |
| 2.2. Syntax | 5 |
| 2.3. Parameters | 5 |
| 2.4. Examples | 9 |
| 2.4.1. Meta attribute search | 9 |
| 2.4.2. Viewing search result status | 10 |
| 2.4.3. Viewing search result | 11 |
| 3. OCR (Optical Character Recognition) search | 15 |
| 3.1. Description | 15 |
| 3.2. Syntax | 15 |
| 3.3. Parameters | 15 |
| 3.4. Examples | 17 |
| 3.4.1. OCR search | 17 |
| 3.4.2. Viewing search result status | 18 |
| 3.4.3. Viewing search result | 19 |
| 4. Object Detection from Image | 22 |
| 4.1. Description | 22 |
| 4.2. Syntax | 22 |
| 4.3. Parameters | 22 |
| 4.4. Examples | 23 |
| 4.4.1. Object detected from image | 23 |
| 5. Image library management | 25 |
| 5.1. Description | 25 |
| 5.2. Syntax | 25 |
| 5.3. Parameters | 25 |
| 5.4. Examples | 27 |
| 5.4.1. Viewing image library | 27 |
| 5.4.2. Adding a new group | 28 |
| 5.4.3. Adding a new image | 29 |
| 5.4.4. Updating an action | 30 |
| 5.4.5. Removing an action | 30 |
| 5.4.6. Making a backup and restoring | 31 |
| 6. AI Timeline | 33 |

| | |
|--|----|
| 6.1. Description | 33 |
| 6.2. Syntax | 33 |
| 6.3. Parameters | 33 |
| 6.4. Examples | 34 |
| 6.4.1. AI timeline search for a duration | 34 |
| 7. AI Engine | 36 |
| 7.1. Description | 36 |
| 7.2. Syntax | 36 |
| 7.3. Parameters | 36 |
| 7.4. Examples | 37 |
| 7.4.1. Viewing AI engine stats | 37 |
| 7.4.2. Enabling AI engine | 38 |
| 8. Face Recognition Search | 40 |
| 8.1. Description | 40 |
| 8.2. Syntax | 40 |
| 8.3. Parameters | 40 |
| 8.4. Examples | 43 |
| 8.4.1. Starting search | 43 |
| 8.4.2. Viewing the search result | 44 |

Chapter 1. Overview

1.1. Description

ai.cgi configures AI-related features in the device.

The following submenus are used for AI functionalities:

- **metaattributesearch**: Used to search recordings with metadata attributes.
- **ocrsearch**: Used to search recordings for OCR text.
- **objectdetectfromimage**: Used to detect the requested object type from images.
- **imagelibrary**: Used to manage images on the device that is used for recognition/training.
- **facerecognitionsearch**: Search recording for the provided face image
- **aiengine**: Used to configure AI engine on the device
- **aitimeline**: Used to view AI timeline for the requested time duration

Chapter 2. Metaattributearch

2.1. Description

The **metaattributearch** submenu is used to search recordings with metadata attributes.

NOTE | This submenu is applicable to NVR only

Access level

| Action | NVR |
|---------|------|
| view | User |
| control | User |

2.2. Syntax

```
http://<Device IP>/stw-cgi/ai.cgi?submenu=  
metaattributearch&action=<value> [&<parameter>=<value> ...]
```

2.3. Parameters

| Action | Parameters | Request/ Response | Type/ Value | Description |
|--------|-----------------|----------------------|------------------------------|--|
| view | Type | REQ | <enum> Results, Status | If Type is passed as Status, search status is informed. If Type is passed as Result, search result is provided. |
| | SearchToken | REQ | <string> | Search session token |
| | ResultFromIndex | REQ | <int> | Index from which search results are fetched |
| | ResultFromTime | REQ | <string> | Time from which search results are fetched. Time in UTC format. YYYY-MM-DDTHH:MM::SSZ |
| | ResultToTime | REQ | <string> | Time to which search results are fetched. Time in UTC format. YYYY-MM-DDTHH:MM::SSZ |

| Action | Parameters | Request/ Response | Type/ Value | Description |
|--------|-----------------------|----------------------|--------------------------------------|---|
| | MaxResults | REQ | <int> | Maximum number of search results to return |
| | Status | RES | <enum> Completed, NotCompleted | Search status |
| | TotalResultsFound | RES | <int> | Total results |
| | TotalCount | RES | <int> | Total count of result |
| | TimedOut | RES | <bool> True, False | Asynchronous search timeout. |
| | ResultFromDate | RES | <string> | Start time of search result. Time in UTC format. YYYY-MM-DDTHH:MM::SSZ |
| | ResultToDate | RES | <string> | End time of search result. Time in UTC format. YYYY-MM-DDTHH:MM::SSZ |
| | SearchTokenExpiryTime | RES | <string> | Time when the search token expires Time in UTC format YYYY-MM-DDTHH:MM:SSZ |
| | Result.#.DateTime | RES | <string> | Time in UTC format YYYY-MM-DDTHH:MM:SSZ |
| | Result.#.Channel | RES | <int> | Result channel ID |
| | Result.#.Attributes | RES | <string> | Attribute text |
| | Result.#.ImageURL | RES | <string> | Image URL |
| | Result.#.Resolution | RES | <string> | Image resolution format=width x height |
| | Result.#.BkID | RES | <string> | Bookmark ID |
| | Result.#.ObjectID | RES | <int> | Object ID |
| | Result.#.BoundingBox | RES | <csv> | Bounding box information of the object in the following format: left, top, right, bottom |

| Action | Parameters | Request/ Response | Type/ Value | Description |
|---------|--|----------------------|--|-----------------------------|
| control | Mode | REQ | <enum> Start, Cancel, Renew, Stop | Search mode |
| | ClassType | REQ | <enum> Person, Vehicle, Face | Search type |
| | ChannelIDList | REQ | <csv> | Search channel ID list |
| | OverlappedID | REQ | <int> | Overlapped number |
| | SearchAttributes.Person. Gender | REQ | <csv> Any, Male, Female | Gender type |
| | SearchAttributes.Person. Clothing.Tops.ColorString | REQ | <csv> Any, Black, Gray, White, Red, Orange, Yellow, Green, Blue, Purple | Color of top (clothing) |
| | SearchAttributes.Person. Clothing.Tops.Length | REQ | <csv> Any, Long, Short | Length of top (clothing) |
| | SearchAttributes.Person. Clothing.Bottoms.ColorString | REQ | <csv> Any, Black, Gray, White, Red, Orange, Yellow, Green, Blue, Purple | Color of bottom (clothing) |
| | SearchAttributes.Person. Clothing.Bottoms.Length | REQ | <csv> Any, Long, Short | Length of bottom (clothing) |
| | SearchAttributes.Person. Clothing.Hat | REQ | <csv> Any, Wear, No | Wearing a hat or not |

| Action | Parameters | Request/ Response | Type/ Value | Description |
|--------|---|----------------------|--|--|
| | SearchAttributes.Person. Belonging.Bag | REQ | <csv> Any, Wear, No | Carrying a bag or not |
| | SearchAttributes.Face.Ge nder | REQ | <csv> Any, Male, Female | Gender |
| | SearchAttributes.Face.Ag eType | REQ | <csv> Any, Young, Adult, Middle, Senior | Age group |
| | SearchAttributes.Face.Ha t | REQ | <csv> Any, Wear, No | Wearing a hat or not |
| | SearchAttributes.Face.Op ticals | REQ | <csv> Any, Wear, No | Wearing glasses or not |
| | SearchAttributes.Vehicle. Type | REQ | <csv> Any, Car, Bus, Truck, Motorcycle, Bicycle, Train | Vehicle type |
| | SearchAttributes.Vehicle. ColorString | REQ | <csv> Any, Black, Gray, White, Red, Orange, Yellow, Green, Blue, Purple | Vehicle color |
| | FromDate | REQ | <string> | Time in UTC format YYYY-MM-DDTHH:MM:SSZ |
| | ToDate | REQ | <string> | Time in UTC format YYYY-MM-DDTHH:MM:SSZ |
| | Async | REQ | <bool>+ True, False | Asynchronous search option |
| | WaitTime | REQ | <int> | Timeout second.(Default:60 sec.) |

| Action | Parameters | Request/ Response | Type/ Value | Description |
|--------|----------------|----------------------|----------------|---|
| | SearchToken | RES | <string> | Search token given as response, which can be used for view operation. |
| | TotalCount | RES | <int> | Total count |
| | ResultFromDate | RES | <string> | Result of 'from' date Time in UTC format YYYY-MM-DDTHH:MM:SSZ |
| | ResultToDate | RES | <string> | Result of 'to' date Time in UTC format YYYY-MM-DDTHH:MM:SSZ |

2.4. Examples

2.4.1. Meta attribute search

REQUEST

```
http://<Device IP>/stw-
cgi/ai.cgi?submenu=metaattributesearch&action=control&Mode=Start&Async=True
&ClassType=Person&ChannelIDList=0,1,2,3,63&OverlappedID=-1&FromDate=1970-01-
01T01:02:03Z&ToDate=2021-01-
26T01:02:03Z&SearchAttributes.Person.Gender=Male&SearchAttributes.Person.Clo
thing.Tops.ColorString=Black,Red&SearchAttributes.Person.Clothing.Bottoms.Co
lorString=Gray,White&SearchAttributes.Person.Belonging.Bag=Wear
```

TEXT RESPONSE

```
HTTP/1.0 200 OK
Content-type: text/plain
<Body>
```

```
SearchToken=35833
```

JSON RESPONSE

```
HTTP/1.0 200 OK
Content-type: application/json
```

```
<Body>
```

```
{  
  "SearchToken": "35833"  
}
```

2.4.2. Viewing search result status

REQUEST

```
http://<Device IP>/stw-  
cgi/ai.cgi?submenu=metaattributearch&action=view&Type=Status&SearchToken=  
35833
```

TEXT RESPONSE

```
HTTP/1.0 200 OK  
Content-type: text/plain  
<Body>
```

```
SearchTokenExpiryTime=2019-06-16T00:28:29Z  
Status=Completed  
TotalResultsFound=0  
TotalCount=1547  
TimedOut=False  
ResultFromDate=2019-06-15T05:38:11Z  
ResultToDate=2019-06-16T00:26:49Z
```

JSON RESPONSE

```
HTTP/1.0 200 OK  
Content-type: application/json  
<Body>
```

```
{  
  "SearchTokenExpiryTime": "2019-06-16T00:28:29Z",  
  "Status": "Completed",  
  "TotalResultsFound": 0,  
  "TotalCount": 1547,
```

```
"TimedOut": "False",  
"ResultFromDate": "2019-06-15T05:38:11Z",  
"ResultToDate": "2019-06-16T00:26:49Z"  
}
```

2.4.3. Viewing search result

REQUEST

```
http://<Device IP>/stw-  
cgi/ai.cgi?submenu=metaattributesearch&action=view&Type=Results&ResultFromI  
ndex=1&MaxResults=100&SearchToken=35833
```

TEXT RESPONSE

```
HTTP/1.0 200 OK  
Content-type: text/plain  
<Body>
```

```
SearchTokenExpiryTime=2019-06-15T23:14:45Z  
Status=Completed  
TotalResultsFound=100  
TotalCount=1414  
TimedOut=False  
Result.0.DateTime=2019-06-15T23:13:32Z  
Result.0.Channel=2  
Result.0.Attributes.Person.Gender=Male  
Result.0.Attributes.Person.Clothing.Tops.ColorString=Black  
Result.0.Attributes.Person.Clothing.Bottoms.ColorString=Gray  
Result.0.Attributes.Person.Belonging.Bag=Wear  
Result.0.ImageURL=/stw-  
cgi/ai.cgi?submenu=imageget&action=view&type=metaattributesearch&ID=0000000  
000000000_10_0_2_1560640412_298530  
Result.0.Resolution=208x336  
Result.0.Channel=2  
Result.0.ObjectID=298530  
Result.0.Coordinate_Del=1372,160,1579,497  
Result.0.BoundingBox=-0.285231,-0.851852,-0.17739,-0.539815  
Result.0.BkID=00000000000000000000000000000000  
...  
Result.99.DateTime=2019-06-15T22:32:27Z
```



```
        }
      },
      "ImageURL": "/stw-
cgi/ai.cgi?msubmenu=imageget&action=view&type=metaattributesearch&ID=0000000
0000000000_10_0_2_1560637947_286642",
      "Resolution": "160x256",
      "ObjectID": 286642,
      "Coordinate_Del": "1221,0,1382,262",
      "BoundingBox": [
        {
          "left": -0.363897,
          "top": -1,
          "right": -0.280021,
          "bottom": -0.757407
        }
      ],
      "BkID": "0000000000000000000000000000000000"
    }
  ]
}
```

Chapter 3. OCR (Optical Character Recognition) search

3.1. Description

The **ocrsearch** submenu is used to search a video that matches an input string.

NOTE This submenu is applicable to NVR only

Access level

| Action | NVR |
|---------|------|
| view | User |
| control | User |

3.2. Syntax

```
http://<Device IP>/stw-cgi/ai.cgi?msubmenu=
ocrsearch&action=<value>[&<parameter>=<value>...]
```

3.3. Parameters

| Action | Parameters | Request/ Response | Type/ Value | Description |
|--------|-----------------|----------------------|------------------------------|--|
| view | Type | REQ | <enum> Results, Status | If Type is passed as Status, search status is informed. If Type is passed as Result, search result is provided. |
| | SearchToken | REQ | <string> | Search session token |
| | ResultFromIndex | REQ | <int> | Index from which search results are fetched |
| | ResultFromTime | REQ | <string> | Time from which search results are fetched. Time in UTC format. YYYY-MM-DDTHH:MM::SSZ |

| Action | Parameters | Request/ Response | Type/ Value | Description |
|--------|-----------------------|----------------------|-----------------------|---|
| | ResultToTime | REQ | <string> | Time to which search results are fetched. Time in UTC format. YYYY-MM-DDTHH:MM::SSZ |
| | MaxResults | REQ | <int> | Maximum number of search results to return |
| | Status | RES | <enum> | Search status |
| | TotalResultsFound | RES | <int> | Total results |
| | TotalCount | RES | <int> | Total result count |
| | TimedOut | RES | <bool> True, False | Asynchronous search timeout. |
| | ResultFromDate | RES | <string> | Start time of search result. Time in UTC format. YYYY-MM-DDTHH:MM::SSZ |
| | ResultToDate | RES | <string> | End time of search result. Time in UTC format. YYYY-MM-DDTHH:MM::SSZ |
| | SearchTokenExpiryTime | RES | <string> | Time when the search token expires Time in UTC format YYYY-MM-DDTHH:MM:SSZ |
| | Result.#.DateTime | RES | <string> | Time in UTC format YYYY-MM-DDTHH:MM:SSZ |
| | Result.#.Channel | RES | <int> | Result channel ID |
| | Result.#.Text | RES | <string> | Text found matching the search keyword |
| | Result.#.ImageURL | RES | <string> | Image URL |
| | Result.#.Resolution | RES | <string> | Image resolution |
| | Result.#.BkID | RES | <string> | Bookmark ID |

| Action | Parameters | Request/Response | Type/Value | Description |
|---------|----------------------|------------------|---|---|
| | Result.#.BoundingBox | RES | <csv> | Bounding box information of the object in the following format: left, top, right, bottom |
| control | Mode | REQ | <enum> Start, Cancel, Renew, Stop | Used to start, cancel, or renew search |
| | ChannelIDList | REQ | <csv> | On which channel search has to be performed |
| | OverlappedID | REQ | <int> | Recording overlapped id |
| | SearchText | REQ | <string> | Text to search in the recording |
| | FromDate | REQ | <string> | Time in UTC format YYYY-MM-DDTHH:MM:SSZ |
| | ToDate | REQ | <string> | Time in UTC format YYYY-MM-DDTHH:MM:SSZ |
| | Async | REQ | <bool>+ True, False | Asynchronous search option |
| | WaitTime | REQ | <int> | Timeout second.(Default:60 sec.) |
| | SearchToken | RES | <string> | Search token given as response, which can be used for view operation. |
| | TotalCount | RES | <int> | Total result count |
| | ResultFromDate | RES | <string> | Time in UTC format YYYY-MM-DDTHH:MM:SSZ |
| | ResultToDate | RES | <string> | Time in UTC format YYYY-MM-DDTHH:MM:SSZ |

3.4. Examples

3.4.1. OCR search

REQUEST

```
http://<Device IP>/stw-  
cgi/ai.cgi?submenu=ocrsearch&action=control&Mode=Start&Async=True&ChannelID  
List=0,1,2,3,4&OverlappedID=-1&FromDate=1970-01-01T01-02-03&ToDate=2021-01-
```

```
01T01-02-03&SearchText=*nu*
```

TEXT RESPONSE

```
HTTP/1.0 200 OK  
Content-type: text/plain  
<Body>
```

```
SearchToken=48928
```

JSON RESPONSE

```
HTTP/1.0 200 OK  
Content-type: application/json  
<Body>
```

```
{  
  "SearchToken": "48928",  
}
```

3.4.2. Viewing search result status

REQUEST

```
http://<Device IP>/stw-  
cgi/ai.cgi?msubmenu=ocrsearch&action=view&Type=Status&SearchToken=48928
```

TEXT RESPONSE

```
HTTP/1.0 200 OK  
Content-type: text/plain  
<Body>
```

```
SearchTokenExpiryTime=  
Status=Completed  
TotalResultsFound=0  
TotalCount=68  
TimedOut=False  
ResultFromDate=2019-06-15T05:43:29Z
```

```
ResultToDate=2019-06-16T00:27:06Z
```

JSON RESPONSE

```
HTTP/1.0 200 OK
Content-type: application/json
<Body>
```

```
{
  "SearchTokenExpiryTime": "",
  "Status": "Completed",
  "TotalResultsFound": 0,
  "TotalCount": 68,
  "TimedOut": "False",
  "ResultFromDate": "2019-06-15T05:43:29Z",
  "ResultToDate": "2019-06-16T00:27:06Z"
}
```

3.4.3. Viewing search result

REQUEST

```
http://<Device IP>/stw-
cgi/ai.cgi?submenu=ocrsearch&action=view&Type=Results&ResultFromIndex=1&Max
Results=5&SearchToken=48928
```

TEXT RESPONSE

```
HTTP/1.0 200 OK
Content-type: text/plain
<Body>
```

```
SearchTokenExpiryTime=2019-06-15T23:21:12Z
Status=Completed
TotalResultsFound=51
TotalCount=55
TimedOut=False
Result.0.DateTime=2019-06-15 23:17:07
Result.0.Channel=0
Result.0.Text=ulnu8
```

```

Result.0.ImageURL=/stw-
cgi/ai.cgi?msubmenu=imageget&action=view&type=ocrsearch&ID=0000000000000000_
10_4_0_1560640627_3078
Result.0.Resolution=96x40
Result.0.BoundingBox=-0.534375,-0.0296296,-0.429167,0.0518519
Result.0.BkID=00000000000000000000000000000000
...
Result.50.DateTime=2019-06-15 06:01:17
Result.50.Channel=0
Result.50.Text=ulnu8
Result.50.ImageURL=/stw-
cgi/ai.cgi?msubmenu=imageget&action=view&type=ocrsearch&ID=0000000000000000_
10_4_0_1560578477_194
Result.50.Resolution=72x24
Result.50.BoundingBox=-0.322917,0.172222,-0.246875,0.22963
Result.50.BkID=00000000000000000000000000000000

```

JSON RESPONSE

```

HTTP/1.0 200 OK
Content-type: application/json
<Body>

```

```

{
  "SearchTokenExpiryTime": "2019-06-15T23:21:12Z",
  "Status": "Completed",
  "TotalResultsFound": 51,
  "TotalCount": 55,
  "TimedOut": "False",
  "Results": [
    {
      "Result": 0,
      "DateTime": "2019-06-15 23:17:07",
      "Channel": 0,
      "Text": "ulnu8",
      "ImageURL": "/stw-
cgi/ai.cgi?msubmenu=imageget&action=view&type=ocrsearch&ID=0000000000000000_
10_4_0_1560640627_3078",
      "Resolution": "96x40",
      "BoundingBox": [
        {

```

```

        "left": -0.534375,
        "top": -0.029630,
        "right": -0.429167,
        "bottom": 0.051852
    }
],
"BkID": "00000000000000000000000000000000"
},
...
{
    "Result": 50,
    "DateTime": "2019-06-15 06:01:17",
    "Channel": 0,
    "Text": "ulnu8",
    "ImageURL": "/stw-
cgi/ai.cgi?msubmenu=imageget&action=view&type=ocrsearch&ID=0000000000000000_
10_4_0_1560578477_194",
    "Resolution": "72x24",
    "BoundingBox": [
        {
            "left": -0.322917,
            "top": 0.172222,
            "right": -0.246875,
            "bottom": 0.229630
        }
    ],
    "BkID": "00000000000000000000000000000000"
}
]
}

```

Chapter 4. Object Detection from Image

4.1. Description

The **objectdetectfromimage** submenu is used to detect the requested object type from images.

NOTE

This submenu is applicable to NVR only

Access level

| Action | NVR |
|---------|------|
| control | User |

4.2. Syntax

```
http://<Device IP>/stw-cgi/ai.cgi?submenu=  
objectdetectfromimage&action=<value> [&<parameter>=<value>...]
```

4.3. Parameters

| Action | Parameters | Request/ Response | Type/ Value | Description |
|---------|----------------------|----------------------|--------------------------|--|
| control | ObjectType | REQ | <enum> Face | Object type to be detected in the past image |
| | Result.#.TempGroupID | RES | <string> | Temporary group ID |
| | Result.#.TempImageID | RES | <int> | Temporary image ID |
| | Result.#.ImageURL | RES | <int> | Image URL of the detected image |
| | Result.#.Resolution | RES | <string> widthxheight | Resolution of the detected object image |
| | Result.#.Coordinates | RES | <string> | Coordinates of the detected object in the original image |

| Action | Parameters | Request/ Response | Type/ Value | Description |
|--------|------------|----------------------|---|---------------|
| | Response | RES | <enum> Success, Fail (No Face Detected), Fail (Too Many Face Detected), Fail (Unknown) | Response code |

4.4. Examples

4.4.1. Object detected from image

REQUEST

```
http://<Device IP>/stw-  
cgi/ai.cgi?msubmenu=objectdetectfromimage&action=control&ObjectType=Face
```

Image sent as POST Content

Base64(image)

JSON RESPONSE

```
HTTP/1.0 200 OK  
Content-type: application/json  
<Body>
```

```
{  
  "Response": "Success",  
  "Results": [  
    {  
      "Result": 0,  
      "TempGroupID": 1000,  
      "TempImageID": 168,  
      "ImageURL": "/stw-
```



```
cgi/ai.cgi?msubmenu=imageget&action=view&type=objectdetetfromimage&ID=168",
  "Resolution": "64x72",
  "Coordinates": [
    {
      "x": 538,
      "y": 349
    },
    {
      "x": 602,
      "y": 349
    },
    {
      "x": 602,
      "y": 421
    },
    {
      "x": 538,
      "y": 421
    }
  ]
}
```

Chapter 5. Image library management

5.1. Description

The **imagelibrary** submenu is used to manage a collection of images in a device.

NOTE | This submenu is applicable to NVR only

Access level

| Action | NVR |
|---------|------|
| view | User |
| add | User |
| update | User |
| remove | User |
| control | User |

5.2. Syntax

```
http://<Device IP>/stw-cgi/ai.cgi?msubmenu=  
imagelibrary&action=<value>[&<parameter>=<value>...]
```

5.3. Parameters

| Action | Parameters | Request/ Response | Type/ Value | Description |
|--------|--------------------------------|----------------------|----------------|--|
| view | GroupID | REQ,RES | <int> | Group ID |
| | GroupID.#.Type | RES | <enum> Face | Group type |
| | GroupID.#.GroupName | RES | <string> | Group name |
| | GroupID.#.ImageID.#.ImageURL | RES | <string> | Image URL |
| | GroupID.#.ImageID.#.Resolution | RES | <string> | Resolution in format "widthxheight" |
| | GroupID.#.ImageID.#.Name | RES | <string> | Name of image |
| add | GroupID | REQ | <int> | Group ID (if a group ID already exists, a new image will be added to it) |

| Action | Parameters | Request/ Response | Type/ Value | Description |
|--------|-------------|----------------------|--|---|
| | Type | REQ | <enum> Face | Type of image |
| | GroupName | REQ | <string> | Group name |
| | Name | REQ | <string> | Image name |
| | ImageRef | REQ | <string> | Image reference to add in the following format: {MetaType}_{Channel}_{UTC}_{ObjectID} ex) 1_0_45545454_1234 |
| | TempGroupID | REQ | <int> | Temporary group ID from object detection submenu |
| | TempImageID | REQ | <int> | Temporary image ID from object detection submenu |
| | GroupID | RES | <int> | Response group ID |
| | ImageID | RES | <int> | Response image ID |
| | Response | RES | <enum> Success, Fail (No Face Detected), Fail (Too Many Face Detected), Fail (Unknown) | Response message |
| update | GroupID | REQ | <int> | Group ID which needs to be updated |
| | Type | REQ | <enum> Face | Type of image |
| | GroupName | REQ | <string> | New group name |
| | ImageID | REQ | <int> | Image ID |
| | Name | REQ | <string> | New name to be updated |
| remove | GroupID | REQ | <int> | Remove image from group ID |
| | ImageIDList | REQ | <csv> | List of image IDs to be removed |
| | ImageID | REQ | <int> | Image ID to be removed |

| Action | Parameters | Request/Response | Type/Value | Description |
|---------|---------------------|------------------|------------------------------|---|
| control | Function | REQ | <enum> Backup, Restore | Back up or restore images |
| | GroupID | REQ | <int> | Group ID to back up or restore |
| | GroupName | REQ | <string> | Group name to back up or restore |
| | Password | REQ | <string> | Password for the backup file |
| | IsPasswordEncrypted | REQ | <bool> True, False | When password is sent as encrypted text |

5.4. Examples

5.4.1. Viewing image library

REQUEST

```
http://<Device IP>/stw-cgi/ai.cgi?msubmenu=imagelibrary&action=view
```

TEXT RESPONSE

```
HTTP/1.0 200 OK
Content-type: text/plain
<Body>
```

```
Group.0.GroupID=1001
Group.0.GroupName=tete
Group.0.Image.0.ImageID=127
Group.0.Image.0.Name=hgg
Group.0.Image.0.ImageURL=/stw-
cgi/ai.cgi?msubmenu=imageget&action=view&type=imagelibrary&ID=face_1001_127
Group.0.Image.0.Resolution=24x24
Group.1.GroupID=1002
Group.1.GroupName=i
```

JSON RESPONSE

```
HTTP/1.0 200 OK
Content-type: application/json
<Body>
```

```
{
  "Groups": [
    {
      "GroupID": 1001,
      "GroupName": "tete",
      "Images": [
        {
          "ImageID": 127,
          "Name": "hgg",
          "ImageURL": "/stw-
cgi/ai.cgi?msubmenu=imageget&action=view&type=imagelibrary&ID=face_1001_127"
,
          "Resolution": "24x24"
        }
      ]
    },
    {
      "GroupID": 1002,
      "GroupName": "i",
      "Images": []
    }
  ]
}
```

5.4.2. Adding a new group

REQUEST

```
http://<Device IP>/stw-
cgi/ai.cgi?msubmenu=imagelibrary&action=add&GroupName=sample
```

TEXT RESPONSE

```
HTTP/1.0 200 OK
Content-type: text/plain
<Body>
```

```
GroupID=1003
ImageID=0
```

JSON RESPONSE

```
HTTP/1.0 200 OK
Content-type: application/json
<Body>
```

```
{
  "GroupID": 1003,
  "ImageID": 0
}
```

5.4.3. Adding a new image

REQUEST

```
http://<Device IP>/stw-
cgi/ai.cgi?submenu=imagelibrary&action=add&GroupID=1003&TempImageID=180&TempGroupID=1000&Name=sampleImage
```

TEXT RESPONSE

```
HTTP/1.0 200 OK
Content-type: text/plain
<Body>
```

```
GroupID=1003
ImageID=186
```

JSON RESPONSE

```
HTTP/1.0 200 OK
Content-type: application/json
<Body>
```

```
{
  "GroupID": 1003,
  "ImageID": 186
}
```

5.4.4. Updating an action

REQUEST

```
http://<Device IP>/stw-  
cgi/ai.cgi?msubmenu=imagelibrary&action=update&GroupID=1003&GroupName=sample  
2
```

TEXT RESPONSE

```
HTTP/1.0 200 OK  
Content-type: text/plain  
<Body>
```

```
OK
```

JSON RESPONSE

```
HTTP/1.0 200 OK  
Content-type: application/json  
<Body>
```

```
{  
  "Response": "Success"  
}
```

5.4.5. Removing an action

REQUEST

```
http://<Device IP>/stw-  
cgi/ai.cgi?msubmenu=imagelibrary&action=remove&GroupID=1003
```

TEXT RESPONSE

```
HTTP/1.0 200 OK  
Content-type: text/plain  
<Body>
```

```
OK
```

JSON RESPONSE

```
HTTP/1.0 200 OK
Content-type: application/json
<Body>
```

```
{
  "Response": "Success"
}
```

5.4.6. Making a backup and restoring

REQUEST

```
http://<Device IP>/stw-
cgi/ai.cgi?submenu=imagelibrary&action=control&Function=Backup&GroupID=1004
&IsPasswordEncrypted=True&Password=Shd/DVG1Y+vLVpTnSTbj8a0W3xd1e7eGS1M/lGjY/
ju5JB670TET+YvdNxzNog3ZtUS/ssxAPVm2402vCQt4CJFz8MfpzfPf7ucQy3QIaivgvm4pwRM1b
PksdUe4Ec3KpyhV+sJWvmWUP+3Hd0hszt0VEXI7Fd5CtZERiLV0vtTRc5DZ6nuY1Vhe+KSDIzJc4
TOSoWNEA9Sv3yUSmZjWprJ4EnlWuZzs3Fdi7l7Xpruq7StRG50myAsu9v6bpoUVaE/ZdM+V1JfDR
xD0ooxVuQDhFY9N0ek0j/JaV10Jq8rb/A4dBhwr59JvdwKgJwoMmQRp/Rt0+4+hGz4fSiHbPg==
```

TEXT/JSON RESPONSE

```
HTTP/1.0 200 OK
Content-type: text/plain
<Body>
```

```
EncryptFileFormat
```

REQUEST

```
http://<Device IP>/stw-
cgi/ai.cgi?submenu=imagelibrary&action=control&Function=Restore&IsPasswordE
ncrypted=True&Password=P182w74Fk394fSftxjHvBR4V3zcMkqXzcRUW0zrLpOMkcoMSeDduS
bt3NP1TkCUf0JbP/oIPmu42ha84Azj52rJdF3P8jNPCFuHHsiQUAsk0S/5U1Swcuc2lCQkQmMwQz
M4mLgHxS1+ZgUtBFy9FA/PKvgqvOe0m6hLywXT/nHT/Aj+0beDcr5t17rB2n1dWlP5v1cipBbJaA
mmLk1SFwZnhilHJ0gE8mDuX8VmQ4WouCuuvsKjRFYZESLhzYCrjoQFX+xZ0bxrwqrdLuWaoJiWH6
JYr1+o75nsQoWJW7RpiJqbfgl0aJDvjvu0l8/YKtb0+J5m4sH5pdSzQk4e3+Q==&GroupName=te
st
```


File sent in POST Content
<Backup File>

JSON RESPONSE

HTTP/1.0 200 OK
Content-type: text/plain
<Body>

```
SamsungTechwin
Content-type:application/json
{
    "Status": "DownloadAck"
}
--SamsungTechwin
Content-type:application/json
{
    "Progress": 0
}
--SamsungTechwin
Content-type:application/json
{
    "Progress": 100
}
--SamsungTechwin
Content-type:application/json
{
    "Status": "DownloadOK"
}
HTTP/1.1 200 OK
Content-type:application/json;charset=utf-8
```

Chapter 6. AI Timeline

6.1. Description

The **aitimeline** submenu is used to view the AI details for the requested time duration.

NOTE | This submenu is applicable to NVR only

Access level

| Action | NVR |
|--------|------|
| view | User |

6.2. Syntax

```
http://<Device IP>/stw-cgi/ai.cgi?submenu=  
aitimeline&action=<value> [&<parameter>=<value>...]
```

6.3. Parameters

| Action | Parameters | Request/ Response | Type/ Value | Description |
|--------|------------------|----------------------|--|--|
| view | ChannelIDList | REQ | <csv> | Requested channel list |
| | ClassType | REQ | <enum> Person, Vehicle, Face, FaceRecogn ition, LicensePlat e | Class type to search |
| | FromDate | REQ | <string> | 'From' date in UTC format: YYYY-MM-DDTHH:MM:SSZ |
| | ToDate | REQ | <string> | 'To' date in UTC format: YYYY-MM-DDTHH:MM:SSZ |
| | OverlappedID | REQ | <int> | Recording overlapped id |
| | TotalCount | RES | <int> | Total result count |
| | Result.#.Channel | RES | <int> | Result channel number |

| Action | Parameters | Request/ Response | Type/ Value | Description |
|--------|-------------------|----------------------|--|--|
| | Result.#.Type | RES | <enum> Person, Vehicle, Face, FaceRecogn ition, LicensePlat e | Result classification type |
| | Result.#.FromDate | RES | <string> | UTC format YYYY-MM-DDTHH:MM:SSZ |
| | Result.#.ToDate | RES | <string> | UTC format YYYY-MM-DDTHH:MM:SSZ |
| | Result.#.BkID | RES | <string> | Bookmark ID |

6.4. Examples

6.4.1. AI timeline search for a duration

REQUEST

```
http://<Device IP>/stw-  
cgi/ai.cgi?&msubmenu=aitimeline&action=view&ClassType=Face&FromDate=1970-01-  
01T01:02:03Z&ToDate=2021-01-26T01:02:03Z&OverlappedID=-1
```

TEXT RESPONSE

```
HTTP/1.0 200 OK  
Content-type: text/plain  
<Body>
```

```
TotalCount=5000  
Result.0.Channel=12  
Result.0.Type=Face  
Result.0.FromDate=2020-06-12T15:24:43Z  
Result.0.ToDate=2020-06-12T15:24:43Z  
Result.0.BkID=00000000000000000000000000000000  
Result.1.Channel=12  
Result.1.Type=Face
```

```
Result.1.FromDate=2020-06-12T15:24:43Z
Result.1.ToDate=2020-06-12T15:24:43Z
Result.1.BkID=00000000000000000000000000000000
...
```

JSON RESPONSE

```
HTTP/1.0 200 OK
Content-type: application/json
<Body>
```

```
{
  "TotalCount": 5000,
  "Results": [
    {
      "Channel": 12,
      "Type": "Face",
      "FromDate": "2020-06-12T15:24:43Z",
      "ToDate": "2020-06-12T15:24:43Z",
      "BkID": "00000000000000000000000000000000"
    },
    {
      "Channel": 12,
      "Type": "Face",
      "FromDate": "2020-06-12T15:24:43Z",
      "ToDate": "2020-06-12T15:24:43Z",
      "BkID": "00000000000000000000000000000000"
    },
    ...
  ]
}
```

Chapter 7. AI Engine

7.1. Description

The **aiengine** submenu is used to manage and view AI engine stats.

NOTE | This submenu is applicable to NVR only

Access level

| Action | NVR |
|--------|------|
| view | User |
| set | User |

7.2. Syntax

```
http://<Device IP>/stw-cgi/ai.cgi?msubmenu=  
aiengine&action=<value>[&<parameter>=<value>...]
```

7.3. Parameters

| Action | Parameters | Request/ Response | Type/ Value | Description |
|--------|--|----------------------|---|---|
| view | ChannelIDList | REQ | <csv> | Requested channel list |
| | TotalEngineUsage.Object EngineUsage | RES | <int> | Total engine usage rate of object detection |
| | TotalEngineUsage.Recog nitionEngineUsage | RES | <int> | Total engine usage rate of recognition |
| | EngineStatus.Channel.#. CamType | RES | <enum> Unknown, MetaDataC am, AIMetaData Cam, NoneMetaC am | Used to notify if the camera is connected to the channel; determines whether to send metadata or not. |
| | EngineStatus.Channel.#. ObjectEngine | RES | <bool> True, False | Enable or disable state of the object engine |
| | EngineStatus.Channel.#. RecognitionEngine | RES | <bool> True, False | Enable or disable state of the recognition engine |

| Action | Parameters | Request/Response | Type/Value | Description |
|--------|---|------------------|-----------------------|--|
| | EngineStatus.Channel.#.ObjectEngineUsage | RES | <int> | Each usage rate of object detection |
| | EngineStatus.Channel.#.RecognitionEngineUsage | RES | <int> | Each usage rate of recognition |
| | FaceRecognitionAgreementTime | RES | <string> | Face recognition agreement time in UTC format. UTCFormat=YYYY-MM-DDTHH:MM:SSZ |
| set | Channel.#.ObjectEngine | REQ | <bool> True, False | Enable or disable object detection engine |
| | Channel.#.RecognitionEngine | REQ | <bool> True, False | Enable or disable recognition engine |
| | AgreementStatus.FaceRecognition | REQ | <bool> True, False | Agreement to run face recognition algorithm. |

7.4. Examples

7.4.1. Viewing AI engine stats

REQUEST

```
http://<Device IP>/stw-cgi/ai.cgi?&msubmenu=aiengine&action=view
```

TEXT RESPONSE

```
HTTP/1.0 200 OK
Content-type: text/plain
<Body>
```

```
TotalEngineUsage.ObjectEngineUsage=46.875
TotalEngineUsage.RecognitionEngineUsage=6.25
EngineStatus.Channel.0.CamType=Unknown
EngineStatus.Channel.0.ObjectEngine=False
EngineStatus.Channel.0.RecognitionEngine=False
EngineStatus.Channel.0.ObjectEngineUsage=0
EngineStatus.Channel.0.RecognitionEngineUsage=0
EngineStatus.Channel.1.CamType=Unknown
EngineStatus.Channel.1.ObjectEngine=False
```

```
EngineStatus.Channel.1.RecognitionEngine=False
EngineStatus.Channel.1.ObjectEngineUsage=0
EngineStatus.Channel.1.RecognitionEngineUsage=0
...
```

JSON RESPONSE

```
HTTP/1.0 200 OK
Content-type: application/json
<Body>
```

```
{
  "TotalEngineUsage": {
    "ObjectEngineUsage": 46.875000,
    "RecognitionEngineUsage": 6.250000
  },
  "EngineStatus": [
    {
      "Channel": 0,
      "CamType": "Unknown",
      "ObjectEngine": false,
      "RecognitionEngine": false,
      "ObjectEngineUsage": 0,
      "RecognitionEngineUsage": 0
    },
    {
      "Channel": 1,
      "CamType": "Unknown",
      "ObjectEngine": false,
      "RecognitionEngine": false,
      "ObjectEngineUsage": 0,
      "RecognitionEngineUsage": 0
    },
    ...
  ]
}
```

7.4.2. Enabling AI engine

REQUEST

```
http://<Device IP>/stw-  
cgi/ai.cgi?&msubmenu=aiengine&action=set&Channel.0.ObjectEngine=True&Channel  
.0.RecognitionEngine=True
```

TEXT RESPONSE

```
HTTP/1.0 200 OK  
Content-type: text/plain  
<Body>
```

```
OK
```

JSON RESPONSE

```
HTTP/1.0 200 OK  
Content-type: application/json  
<Body>
```

```
{  
  "Response": "Success"  
}
```


Chapter 8. Face Recognition Search

8.1. Description

The **facerecognitionsearch** submenu is used to search face recognition information from the recordings on channels with face recognition enabled.

NOTE This submenu is applicable to NVR only

Access level

| Action | NVR |
|---------|------|
| view | User |
| control | User |
| remove | User |

8.2. Syntax

```
http://<Device IP>/stw-cgi/ai.cgi?submenu=  
facerecognitionsearch&action=<value> [&<parameter>=<value>...]
```

8.3. Parameters

| Action | Parameters | Request/ Response | Type/ Value | Description |
|--------|-----------------|----------------------|--|---|
| view | Type | REQ | <enum> SearchImages, Results, Status | Changes according to the type information provided |
| | SearchToken | REQ | <string> | Search token received in control operation |
| | ResultFromIndex | REQ | <int> | From which result index the results need to be displayed. |
| | ResultFromTime | REQ | <string> | Time from which search results are fetched. Time in UTC format. YYYY-MM-DDTHH:MM::SSZ |

| Action | Parameters | Request/ Response | Type/ Value | Description |
|--------|-----------------------|----------------------|--|---|
| | ResultToTime | REQ | <string> | Time to which search results are fetched. Time in UTC format. YYYY-MM-DDTHH:MM::SSZ |
| | MaxResults | REQ | <int> | Maximum results needed in view |
| | Status | RES | <enum> Completed, NotComple ted | When Type is set to Status, search status is provided |
| | TotalResultsFound | RES | <int> | Total results |
| | TotalCount | RES | <int> | Total count |
| | TimedOut | RES | <bool> True, False | Asynchronous search timeout. |
| | ResultFromDate | RES | <string> | Start time of search result. Time in UTC format. YYYY-MM-DDTHH:MM::SSZ |
| | ResultToDate | RES | <string> | End time of search result. Time in UTC format. YYYY-MM-DDTHH:MM::SSZ |
| | SearchTokenExpiryTime | RES | <string> | Search token expiry date in UTC format YYYY-MM-DDTHH:MM:SSZ |
| | Result.#.DateTime | RES | <string> | Date time in UTC format YYYY-MM-DDTHH:MM:SSZ |
| | Result.#.Channel | RES | <int> | Result channel id |
| | Result.#.ObjectID | RES | <int> | Object id |
| | Result.#.ImageURL | RES | <string> | Image URL can be used to download image |
| | Result.#.Resolution | RES | <string> widthxheight | Result image resolution |
| | Result.#.BkID | RES | <string> | Bookmark ID |

| Action | Parameters | Request/ Response | Type/ Value | Description |
|---------|-----------------------------------|----------------------|---|---|
| | Result.#.BoundingBox | RES | <csv> | Bounding box information in the following format: left, top, right, bottom |
| | Result.#.SearchImage.#.Type | RES | <enum> Library, File | When Type is set to SearchImages |
| | Result.#.SearchImage.#.GroupName | RES | <int> | Group name |
| | Result.#.SearchImage.#.ImageName | RES | <int> | Image name |
| | Result.#.SearchImage.#.ImageURL | RES | <string> | Image URL |
| | Result.#.SearchImage.#.Similarity | RES | <int> | Similarity score 1-100 |
| control | Mode | REQ | <enum> Start, Cancel, Renew, Stop | Search mode |
| | ChannelIDList | REQ | <csv> | Channels on which search is performed |
| | OverlappedID | REQ | <int> | Recording overlapped id |
| | FromDate | REQ | <string> | Time in UTC format "YYYY-MM-DDTHH:MM:SSZ" |
| | ToDate | REQ | <string> | Time in UTC format "YYYY-MM-DDTHH:MM:SSZ" |
| | Async | REQ | <bool>+ True, False | Asynchronous search option |
| | WaitTime | REQ | <int> | Timeout second.(Default:60 sec.) |
| | SearchImage.#.Type | REQ | <enum> Library | Search image type |
| | SearchImage.#.GroupID | REQ | <int> | Group ID from image library |
| | SearchImage.#.ImageID | REQ | <int>□ | Image ID from image library |
| | Similarity | REQ | <int> | Similarity threshold for filtering results above this value |
| | SearchToken | RES | <string> | Search token for the requested search query |

| Action | Parameters | Request/ Response | Type/ Value | Description |
|--------|----------------|----------------------|----------------|---|
| | TotalCount | RES | <int> | Total result count |
| | ResultFromDate | RES | <string> | Result of 'from' date in UTC format YYYY-MM-DDTHH:MM:SSZ |
| | ResultToDate | RES | <string> | Result of 'to' date in UTC format YYYY-MM-DDTHH:MM:SSZ |
| remove | SearchImage | REQ | <int> | Search image index |
| | SearchToken | REQ | <string> | Search token |

8.4. Examples

8.4.1. Starting search

REQUEST

```
http://<Device IP>/stw-
cgi/ai.cgi?submenu=facerecognitionsearch&action=control&Mode=Start&Async=Tr
ue&ChannelIDList=0,1,2,3,4,5,6,7,8&OverlappedID=-1&FromDate=2000-03-
01T10:59:23Z&ToDate=2021-03-
11T11:59:23Z&SearchImage.0.Type=Library&SearchImage.0.GroupID=1001&SearchIma
ge.0.ImageID=3
```

TEXT RESPONSE

```
HTTP/1.0 200 OK
Content-type: text/plain
<Body>
```

```
SearchToken=97923
```

JSON RESPONSE

```
HTTP/1.0 200 OK
Content-type: application/json
<Body>
```

```
{
```

```
"SearchToken": "97923"
}
```

8.4.2. Viewing the search result

REQUEST

```
http://<Device IP>/stw-
cgi/ai.cgi?msubmenu=facerecognitionsearch&action=view&Type=Status&SearchToken=97923
```

TEXT RESPONSE

```
HTTP/1.0 200 OK
Content-type: text/plain
<Body>
```

```
Status=Completed
TotalResultsFound=0
TotalCount=48
TimedOut=False
SearchTokenExpiryTime=2020-06-15T19:30:13Z
ResultFromDate=2020-06-12T11:34:41Z
ResultToDate=2020-06-12T15:24:43Z
```

JSON RESPONSE

```
HTTP/1.0 200 OK
Content-type: application/json
<Body>
```

```
{
  "Status": "Completed",
  "TotalResultsFound": 0,
  "TotalCount": 48,
  "TimedOut": "False",
  "SearchTokenExpiryTime": "2020-06-15T19:26:12Z"
  "ResultFromDate": "2020-06-12T11:34:41Z"
  "ResultToDate": "2020-06-12T15:24:43Z"
```

```
}
```

REQUEST

```
http://<Device IP>/stw-  
cgi/ai.cgi?msubmenu=facerecognitionsearch&action=view&Type=Results&ResultFromIndex=1&MaxResults=100&SearchToken=97923
```

TEXT RESPONSE

```
HTTP/1.0 200 OK  
Content-type: text/plain  
<Body>
```

```
Status=Completed  
TotalResultsFound=48  
TotalCount=48  
TimedOut=False  
SearchTokenExpiryTime=2020-06-16T08:13:57Z  
Result.0.DateTime=2020-06-12T15:24:43Z  
Result.0.Channel=2  
Result.0.ObjectID=3745  
Result.0.ImageURL=/stw-  
cgi/ai.cgi?msubmenu=imageget&action=view&type=frsearch&ID=000918E1A14D0000_1  
0_1_2_1591975483_3745  
Result.0.Resolution=288x392  
Result.0.BoundingBox=-1,-0.261111,-0.848398,0.103704  
Result.0.BkID=00000000000000000000000000000000  
Result.0.SearchImage.0.Type=Library  
Result.0.SearchImage.0.GroupName=sample2  
Result.0.SearchImage.0.ImageName=sampleImage  
Result.0.SearchImage.0.ImageURL=/stw-  
cgi/ai.cgi?msubmenu=imageget&action=view&type=imagelibrary&ID=face_1003_186  
Result.0.SearchImage.0.Similarity=54  
...  
Result.47.DateTime=2020-06-12T11:34:41Z  
Result.47.Channel=3  
Result.47.ObjectID=63  
Result.47.ImageURL=/stw-  
cgi/ai.cgi?msubmenu=imageget&action=view&type=frsearch&ID=000918E1A14D0000_1  
0_1_3_1591961681_63
```

```
.Result.47.Resolution=80x72
Result.47.BoundingBox=-0.952083,-0.131481,-0.861458,0.0111111
Result.47.BkID=00000000000000000000000000000000
Result.47.SearchImage.0.Type=Library
Result.47.SearchImage.0.GroupName=sample2
Result.47.SearchImage.0.ImageName=sampleImage
Result.47.SearchImage.0.ImageURL=/stw-
cgi/ai.cgi?msubmenu=imageget&action=view&type=imagelibrary&ID=face_1003_186
Result.47.SearchImage.0.Similarity=54
```

JSON RESPONSE

```
HTTP/1.0 200 OK
Content-type: application/json
<Body>
```

```
{
  "Status": "Completed",
  "TotalResultsFound": 48,
  "TotalCount": 48,
  "TimedOut": "False",
  "SearchTokenExpiryTime": "2020-06-16T08:14:07Z",
  "Results": [
    {
      "DateTime": "2020-06-12T15:24:43Z",
      "Channel": 2,
      "ObjectID": 3745,
      "ImageURL": "/stw-
cgi/ai.cgi?msubmenu=imageget&action=view&type=frsearch&ID=000918E1A14D0000_1
0_1_2_1591975483_3745",
      "Resolution": "288x392",
      "BoundingBox": [
        {
          "left": -1,
          "top": -0.261111,
          "right": -0.848398,
          "bottom": 0.103704
        }
      ],
      "BkID": "00000000000000000000000000000000",
      "SearchImages": [
```

```

        {
            "Type": "Library",
            "GroupName": "sample2",
            "ImageName": "sampleImage",
            "ImageURL": "/stw-
cgi/ai.cgi?msubmenu=imageget&action=view&type=imagelibrary&ID=face_1003_186"
        ,
            "Similarity": 54
        }
    ]
},
...
{
    "DateTime": "2020-06-12T11:34:41Z",
    "Channel": 3,
    "ObjectID": 63,
    "ImageURL": "/stw-
cgi/ai.cgi?msubmenu=imageget&action=view&type=frsearch&ID=000918E1A14D0000_1
0_1_3_1591961681_63",
    "Resolution": "80x72",
    "BoundingBox": [
        {
            "left": -0.952083,
            "top": -0.131481,
            "right": -0.861458,
            "bottom": 0.011111
        }
    ],
    "BkID": "00000000000000000000000000000000",
    "SearchImages": [
        {
            "Type": "Library",
            "GroupName": "sample2",
            "ImageName": "sampleImage",
            "ImageURL": "/stw-
cgi/ai.cgi?msubmenu=imageget&action=view&type=imagelibrary&ID=face_1003_186"
        ,
            "Similarity": 54
        }
    ]
}

```


]

}