

IBM Visual Insights

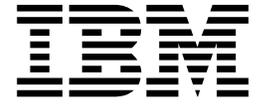


# Manual del usuario

*Versión 1 Release 1*



IBM Visual Insights



# Manual del usuario

*Versión 1 Release 1*



---

## Nota

Antes de utilizar esta información y el producto al que da soporte, lea la información del apartado “Avisos” en la página 29.



---

Primera edición (septiembre de 2017)



---

# Contenido

## Capítulo 1. Visión general del producto 1

Roles . . . . .	1
¿Cómo se hace una copia de seguridad y una restauración de los datos? . . . . .	1
Novedades de este release . . . . .	2
Características de accesibilidad . . . . .	2

## Capítulo 2. Creación de sistemas perimetrales . . . . . 5

Requisitos de los sistemas perimetrales . . . . .	5
Instalación de paquetes de NVIDIA GPU . . . . .	5
Instalación de Caffe . . . . .	6
Resolución de problemas de la instalación de Caffe . . . . .	7
Instalación de Open CV . . . . .	8
Instalación de bibliotecas de detección de objetos . . . . .	8
Configuración del servidor de imágenes . . . . .	10
Configuración del almacén de modelos . . . . .	10
Registro del sistema perimetral en la aplicación central . . . . .	11

## Capítulo 3. Creación y uso de modelos 13

Estructura de archivos de imagen comprimidos . . . . .	13
--	----

Adición de imágenes históricas para grupos de imágenes . . . . .	14
Creación de solicitudes de modelos . . . . .	15
Modelos formados . . . . .	16
Estructura de los archivos de modelo. . . . .	16
Modelos validados . . . . .	21
Distribución de modelos formados en sistemas perimetrales . . . . .	22
Modelos de repetición de formación . . . . .	22

## Capítulo 4. Comprobación de los resultados de la inspección. . . . . 25

Imágenes . . . . .	25
Filtrado de defectos. . . . .	25
Comprobación de defectos . . . . .	25

## Capítulo 5. Panel de control del indicador clave de rendimiento . . . . . 27



---

## Capítulo 1. Visión general del producto

IBM® Visual Insights es una solución de alertas y supervisión de la calidad de fabricación que puede tomar imágenes de productos y ensamblajes, tanto en curso como finalizados, y clasificarlas en categorías de defectos.

---

### Roles

Para comprender Visual Insights, resulta útil comprender cómo interactúan los distintos roles con el producto.

*Tabla 1. Roles de Visual Insights*

Rol	Descripción
Administrador de modelos	Gestiona modelos y tipos de defectos, carga conjuntos de imágenes para tipos de defectos específicos, crea solicitudes de formación de modelos para el Experto en Datos y distribuye modelos ejecutables a sistemas perimetrales.
Experto en datos	Da formación a los modelos en función de los conjuntos de imágenes y los tipos de defectos creados por el Administrador de modelos. La formación de los modelos se ejecuta fuera de Visual Insights. Es aconsejable que el Experto en datos utilice el sistema DIGITS (Deep Learning GPU Training System) de NVIDIA para dar formación a los modelos.
Inspector	Verifica los resultados de la inspección que produce el producto, cambia los tipos de defectos si es necesario, señala los tipos de defectos desconocidos y los envía al Supervisor de inspectores para su evaluación adicional.
Supervisor de inspectores	Vuelve a comprobar los resultados de la inspección del Inspector. Revisa y clasifica los tipos de defectos desconocidos. Revisa el panel de control del indicador clave de rendimiento, que incluye el total de defectos, la tasa de defectos, la sobresaturación y las tasas de defectos de escape.

---

### ¿Cómo se hace una copia de seguridad y una restauración de los datos?

La redundancia de IBM Open Platform se utiliza para proteger los datos del cliente un entorno de Big Data. Además, Tivoli Storage Manager se utiliza para la copia de seguridad de datos en los entornos de producción, lo que incluye los archivos de Linux. Los archivos Linux incluyen archivos de cliente cargados y archivos de configuración y registro de middleware y aplicaciones.

En la tabla siguiente se muestra la planificación de copia de seguridad para diversos aspectos de la solución.

Tabla 2. Planificación de copia de seguridad para los datos de la solución

Datos	Tipo de copia de seguridad	Frecuencia	Hora (Hora central)	Periodo de retención
Archivos	Completa	Quincenal	00:00 - 03:00 el día 1 y el día 16 de cada mes	5 semanas
Archivos	Incremental	Dos veces al día	Entre las 00:00 - 03:00 y las 12:00 - 15:00	14 días

Durante el intervalo de copia de seguridad, se puede acceder a la solución. Sin embargo, es posible que el rendimiento se vea afectado.

En caso de que se produzca un error del sistema que provoque daños o pérdida de los datos, IBM le ayudará a restaurar los datos en los puntos de tiempo de recuperación, según su política de copia de seguridad.

---

## Novidades de este release

Las siguientes nuevas características están disponibles en IBM Visual Insights.

### Nuevas características y mejoras

- Ahora puede utilizar grupos de imágenes para representar el mismo tipo de imágenes utilizando uno o varios archivos de imágenes comprimidos.
- Se ha añadido soporte para varias versiones de modelo que comparten los mismos grupos de imágenes, pero utilizan distintos archivos de imágenes para dar formación al modelo.
- Se ha añadido el proceso de repetición de formación de modelos. Puede repetir la formación manual o automáticamente de una nueva versión de modelo utilizando distintos archivos de imágenes.
- Se ha añadido soporte para la validación de modelos. El proceso de validación calcula y muestra un informe de precisión de modelo basado en los archivos de imágenes de validación.
- Ahora puede utilizar recuadros de defectos y tipos de defectos en una imagen para marcar la ubicación del defecto para el inspector.
- Ahora puede mostrar varias ubicaciones de defectos en una imagen. Puede añadir, ajustar y suprimir recuadros de defectos en una imagen.
- Se ha actualizado el panel de control de KPI para incluir los defectos por unidad y la tasa de defectos para el supervisor de inspectores.
- Se ha añadido soporte para que el modelo de detección de objetos pueda detectar varios defectos en una imagen con el modelo CNN.

---

## Características de accesibilidad

Las características de accesibilidad ayudan a los usuarios que tienen una discapacidad física, por ejemplo movilidad restringida o visión limitada, a utilizar productos de tecnología de la información.

Para obtener información acerca del compromiso que IBM tiene con la accesibilidad, visite el IBM Accessibility Center ([www.ibm.com/able](http://www.ibm.com/able)).

La documentación HTML tiene funciones de accesibilidad. Los documentos PDF son suplementarios y, como tales, no incluyen características de accesibilidad adicionales.



---

## Capítulo 2. Creación de sistemas perimetrales

Visual Insights consta de la aplicación central y los perimetrales. Los sistemas perimetrales son sistemas de Linux utilizados para ejecutar una detección de defectos en tiempo de ejecución.

Los sistemas perimetrales utilizan la infraestructura de aprendizaje profundo de Caffe. Caffe es un entorno de formación de red neuronal artificial (ANN) dedicado. El aprendizaje profundo requiere una cantidad significativa de recursos de procesamiento. El aprendizaje profundo se puede ejecutar de forma eficaz utilizando una unidad de procesamiento gráfico (GPU). Aunque la mayoría de las infraestructuras de aprendizaje profundo son también compatibles con el procesamiento de la CPU, el procesamiento de la GPU ofrece un rendimiento aceptable en entornos de producción.

---

### Requisitos de los sistemas perimetrales

Antes de crear un sistema perimetral, asegúrese de que el sistema cumpla los requisitos.

- Ubuntu 16.04
- procesador de 4 núcleos
- 64 GB de memoria
- unidad de disco duro de 2 TB
- Una o varias tarjetas NVIDIA GPU

---

### Instalación de paquetes de NVIDIA GPU

Para habilitar el procesamiento de la GPU, hay que instalar los paquetes de NVIDIA GPU necesarios.

#### Procedimiento

1. Descargue e instale los controladores de NVIDIA GPU. La lista de controladores de NVIDIA para Ubuntu está disponible en el enlace siguiente: [Binary Driver How to - Nvidia](#).
2. Descargue el kit de herramientas NVIDIA CUDA 8.0 desde el enlace siguiente: [Descargas de CUDA 8.0](#) En el sitio web de descargas, elija **Linux x86\_64 Ubuntu 16.04 deb (network)** como plataforma de destino. Se descarga un archivo de instalación `cuda-repo-ubuntu1604_8.0.61-1_amd64.deb`.
3. Utilice los mandatos siguientes para instalar el archivo de CUDA en el servidor de destino:

```
sudo dpkg -i cuda-repo-ubuntu1604_8.0.61-1_amd64.deb
sudo apt-get update
sudo apt-get install cuda
```
4. Descargue la biblioteca CUDNN-v5.1 de NVIDIA CUDA Deep Neural Network desde el enlace siguiente: [NVIDIA cuDNN](#). Es necesario que se registre antes de la descarga.
5. Desempaquete el archivo `cuda-repo-ubuntu1604_8.0.61-1_amd64.deb` descargado en el directorio de instalación de cuda utilizando el mandato siguiente:

```
sudo tar -xvf cudnn-8.0-linux-x64-v5.1.tgz -C /usr/local
```

6. Utilice el mandato siguiente para establecer la variable de entorno:  
`export LD_LIBRARY_PATH=/usr/local/cuda/lib64:$LD_LIBRARY_PATH`  
 Añada también este mandato al script `~/ .bashrc`.
7. Utilice los mandatos siguientes para instalar el paquete NCCL de NVIDIA:  
`git clone https://github.com/NVIDIA/nvcl.git`  
`cd nvcl`  
`sudo make install -j4`

---

## Instalación de Caffe

Antes de utilizar un sistema perimetral, tiene que instalar la infraestructura de aprendizaje profundo de Caffe y los paquetes relacionados. Caffe se utiliza para la formación de modelos y la clasificación de defectos.

### Procedimiento

1. Utilice los mandatos siguientes para instalar los paquetes necesarios para Caffe:  
`sudo apt-get update`  
`sudo apt-get upgrade`  
`sudo apt-get install -y build-essential cmake git pkg-config`  
`sudo apt-get install -y libprotobuf-dev libleveldb-dev libsnappy-dev`  
`libhdf5-serial-dev protobuf-compiler`  
`sudo apt-get install -y libatlas-base-dev libjasper-dev`  
`sudo apt-get install -y --no-install-recommends libboost-all-dev`  
`sudo apt-get install -y libgflags-dev libgoogle-glog-dev liblmdb-dev`  
`sudo apt-get install -y python-pip`  
`sudo apt-get install -y python-dev`  
`sudo apt-get install -y python-numpy python-scipy`  
`sudo apt-get install -y libopencv-dev`
2. Utilice el mandato siguiente para descargar el código fuente de Caffe:  
`wget https://github.com/BVLC/caffe/archive/rc5.zip`
3. Utilice los mandatos siguientes para desempaquetar el paquete y especificar el directorio del paquete:  
`unzip rc5.zip`  
`cd ./caffe-rc5`
4. Utilice el mandato siguiente para realizar una copia del archivo de configuración make:  
`cp Makefile.config.example Makefile.config`
5. Añada las variables siguientes en el archivo `Makefile.config`:  
`USE_CUDNN := 1`  
`CUDA_DIR := /usr/local/cuda-8.0`  
`PYTHON_INCLUDE := /usr/include/python2.7 \`  
`/usr/lib/python2.7/dist-packages/numpy/core/include`  
`PYTHON_LIB := /usr/lib/x86_64-linux-gnu`  
`WITH_PYTHON_LAYER := 1`  
`INCLUDE_DIRS := $(PYTHON_INCLUDE) /usr/local/include`  
`/usr/include/hdf5/serial`  
`LIBRARY_DIRS := $(PYTHON_LIB) /usr/local/lib /usr/lib`  
`/usr/lib/x86_64-linux-gnu /usr/lib/x86_64-linux-gnu/hdf5/serial`
6. En el directorio `caffe-rc5`, ejecute el mandato siguiente:  
`find . -type f -exec sed -i -e 's^"hdf5.h"^"hdf5/serial/hdf5.h"^g' -e`  
`'s^"hdf5_h1.h"^"hdf5/serial/hdf5_h1.h"^g' '{} ' \;`
7. Ejecute los mandatos siguientes:  
`cd /usr/lib/x86_64-linux-gnu`

- ```
sudo ln -s libhdf5_serial.so.10.1.0 libhdf5.so
sudo ln -s libhdf5_serial_hl.so.10.0.2 libhdf5_hl.so
```
8. Utilice los mandatos siguientes para instalar los paquetes de Python necesarios en el directorio `caffe-rc5/python`:
 

```
cd {ruta_instalación_caffe}/caffe-rc5/python
for req in $(cat requirements.txt); do sudo -H pip install $req
--upgrade; done
```

 donde `{ruta_instalación_caffe}` es la ruta de despliegue de Caffe.
  9. Abra el archivo `make` en el directorio `{ruta_instalación_caffe}` y cambie el parámetro `NVCCFLAGS` al valor siguiente:
 

```
NVCCFLAGS += -D_FORCE_INLINES -ccbin=$(CXX) -Xcompiler -fPIC
$(COMMON_FLAGS)
```
  10. En el directorio principal de Caffe `caffe-rc5`, utilice los mandatos siguientes para iniciar la creación y la instalación de Caffe:
 

```
make all
make test
make runtest
make pycaffe
make distribute
```
  11. Añada la línea siguiente al script `~/bashrc`:
 

```
export PYTHONPATH="/usr/lib/python2.7:{ruta_instalación_caffe}/caffe-rc5/python:$PYTHONPATH"
```

 donde `{ruta_instalación_caffe}` es la ruta de despliegue de Caffe.

## Resolución de problemas de la instalación de Caffe

Si aparece un mensaje de error en el registro cuando inicia la compilación y la instalación de Caffe, puede seguir estos pasos para intentar resolver el problema.

### Síntomas

Cuando inicia la compilación y la instalación de Caffe, aparece el siguiente mensaje:

```
1. In file included from ./include/caffe/util/device_alternate.hpp:40:0,
2.     from ./include/caffe/common.hpp:19,
3.     from src/caffe/common.cpp:7:
4. ./include/caffe/util/cudnn.hpp: In function 'void caffe::cudnn::createPoolingDesc(cudnnPoolingStruct**,
caffe::PoolingParameter_PoolMethod, cudnnPoolingMode_t*, int, int, int, int, int)':
5. ./include/caffe/util/cudnn.hpp:127:41: error: too few arguments to function 'cudnnStatus_t
cudnnSetPooling2dDescriptor(cudnnPoolingDescriptor_t, cudnnPoolingMode_t, cudnnNanPropagation_t, int,
int, int, int, int, int)'
6.     pad_h, pad_w, stride_h, stride_w));
7.     ^
8. ./include/caffe/util/cudnn.hpp:15:28: note: in definition of macro 'CUDNN_CHECK'
9.     cudnnStatus_t status = condition; \
10.     ^
11. In file included from ./include/caffe/util/cudnn.hpp:5:0,
12.     from ./include/caffe/util/device_alternate.hpp:40,
13.     from ./include/caffe/common.hpp:19,
14.     from src/caffe/common.cpp:7:
15. /usr/local/cuda-7.5//include/cudnn.h:803:27: note: declared here
16.     cudnnStatus_t CUDNNWINAPI cudnnSetPooling2dDescriptor(
17.     ^
18. make: *** [.build_release/src/caffe/common.o] Error 1
19.
```

### Resolución del problema

Para solucionar el error, consulte los siguientes pasos:

1. En el directorio `/include/caffe/util/cudnn.hpp`, sustituya el archivo `cudnn.hpp` por el archivo `cudnn.hpp` más reciente, que está en el repositorio de Caffe en GitHub.
2. En la carpeta `/src/caffe/layers`, sustituya todos los archivos `cudnn` que hay en la carpeta `/src/caffe/layers` por los archivos `cudnn` más recientes, que están en el repositorio de Caffe en GitHub.

---

## Instalación de Open CV

Antes de utilizar un sistema perimetral, tiene que instalar la biblioteca de Open Source Computer Vision (OpenCV).

### Procedimiento

1. Obtenga el código fuente de OpenCV de Github:  
`wget https://github.com/opencv/opencv/archive/3.2.0.zip`
2. Desempaque el paquete descargado y cambie al directorio del paquete:  
`unzip 3.2.0.zip`  
`cd opencv-3.2.0`
3. Cree un subdirectorio de compilación y cambie al directorio:  
`mkdir build`  
`cd build`
4. Prepare y genere la configuración de compilación:  
`cmake -D CMAKE_BUILD_TYPE=RELEASE -D CMAKE_INSTALL_PREFIX=/usr/local -D WITH_TBB=ON -D WITH_V4L=ON ..`
5. Compile y cree el paquete:  
`make -j $(nproc)`
6. Instale el paquete:  
`sudo make install`
7. Registre las bibliotecas y módulos para el sistema:  
`sudo /bin/bash -c 'echo "/usr/local/lib" > /etc/ld.so.conf.d/opencv.conf'`  
`sudo ldconfig`
8. Si es necesario, desinstale la versión de opencv antigua para evitar el conflicto de versión:  
`sudo apt-get autoremove libopencv-dev`

---

## Instalación de bibliotecas de detección de objetos

Debe instalar una biblioteca de detección de objetos para poder ejecutar el modelo de detección de objetos en un sistema perimetral.

### Acerca de esta tarea

IBM Visual Insights da soporte a las siguientes bibliotecas de detección de objetos: YOLO (You Only Look Once), Faster R-CNN y SSD (Single Shot MultiBox Detector).

### Procedimiento

1. Utilice los mandatos siguientes para instalar los paquetes de Python relacionados:  
`sudo apt-get install python-numpy`  
`sudo apt-get install python-scipy`  
`pip install cython`

```

pip install eaydict
pip install shutil
pip install cPickle
pip install uuid
pip install multiprocessing
pip install xml

```

2. Instale una de las siguientes bibliotecas:

| Biblioteca                           | Instrucciones de instalación                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
|--------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Biblioteca YOLO versión 2</b>     | <ol style="list-style-type: none"> <li>Ejecute el siguiente mandato para obtener el código fuente de YOLO:<br/><b>git clone --recursive http://github.com/pjreddie/darknet.git</b></li> <li>Editar el archivo Makefile habilitando los parámetros <b>GPU</b> y <b>CUDNN</b>, y seleccione el parámetro <b>GPU ARCH</b> correcto según su configuración de máquina.</li> <li>Ejecute el siguiente comando para compilar YOLO:<br/><b>make</b></li> </ol>                                                                                                                                              |
| <b>Biblioteca Python Faster-RCNN</b> | <ol style="list-style-type: none"> <li>Ejecute el siguiente mandato para obtener el código fuente de Faster R-CNN:<br/><b>git clone --recursive http://github.com/rbgirshick/py-faster-rcnn.git</b></li> <li>En el directorio py-faster-rcnn, en la carpeta lib, ejecute el siguiente mandato para compilar Cython:<br/><b>make</b></li> <li>Compile Caffe y pycaffe dentro de la carpeta py-faster-rcnn. Debe compilar Caffe utilizando la capa de Python.</li> </ol>                                                                                                                               |
| <b>Biblioteca SSD</b>                | <ol style="list-style-type: none"> <li>Ejecute el siguiente mandato para obtener el código fuente de SSD:<br/><b>git clone --recursive https://github.com/weiliu89/caffe.git</b></li> <li>Edite el archivo Makefile y cambie los parámetros <b>CUDA_ARCH</b>, <b>BLAS</b> y <b>PYTHON_INCLUDE</b> según su configuración de máquina.</li> <li>Compile el código con el siguiente mandato:<br/><b>make -j8</b></li> <li>Compile la capa de Python utilizando el siguiente mandato:<br/><b>make py</b></li> <li>Compile la prueba utilizando el siguiente mandato:<br/><b>make test -j8</b></li> </ol> |

- Añadir las siguientes variables de entorno a nivel de usuario: *YOLO\_HOME*, *FRCNN\_HOME* y *SSD\_HOME*. El siguiente texto es un ejemplo de adición de variables de entorno: *YOLO\_HOME=/home/user/darknet/*, *FRCNN\_HOME=/home/user/py-faster-rcnn/*.
- Opcional: Despliegue manualmente la biblioteca de detección de objetos.
  - Desempaquete el paquete de detección de objetos.
  - Añada el archivo *yolo/detectorobj.c* a la carpeta *darknet/examples*.
  - Edite el archivo *darknet/Makefile* e indique *EXECOBJA=detectorobj.o*. El siguiente código es un ejemplo de código en el archivo *Makefile*:  

```

EXECOBJA=detectorobj.o captcha.o lsd.o super.o voxel.o art.o tag.o
cifar.o go.o rnn.o rnn_vid.o compare.o segmenter.o regressor.o
classifier.o coco.o dice.o yolo.o detector.o writing.o nightmare.o
swag.o darknet.o

```

- d. En el archivo `Makefile`, añada `$(EXECOBJ)` para los objetos `$(SLIB)` y `$(ALIB)`. El siguiente código es un ejemplo de código en el archivo `Makefile`:

```
$(ALIB): $(EXECOBJ) $(OBJS)
$(AR) $(ARFLAGS) $@ $^
$(SLIB): $(EXECOBJ) $(OBJS)
$(CC) $(CFLAGS) -shared $^ -o $@ $(LDFLAGS)
```
- e. Ejecute el mandato siguiente:  
**make**
- f. Copie el archivo `iotmyolo.py` y añádalo al directorio `YOLO_HOME`.

---

## Configuración del servidor de imágenes

El usuario configura el servidor de imágenes en una máquina perimetral, de tal manera que la máquina pueda almacenar las imágenes capturadas por una cámara industrial. El controlador del sistema perimetral es quien supervisa al servidor de imágenes. Cuando se añade una nueva imagen, se puntúa, y el resultado de la inspección se envía a la aplicación central para que un inspector pueda evaluar la imagen y los resultados de la inspección.

### Antes de empezar

Añada la línea siguiente al archivo de exportación:  
`/imageserver "dirección IP"(rw, sync, no_root_squash, no_all_squash)`. Por ejemplo, añadiría la siguiente línea al archivo:  
`/imageserver 10.173.0.0/29 (rw, sync, no_root_squash, no_all_squash)`.

### Procedimiento

1. Utilice el mandato siguiente para instalar e iniciar el servicio `nfs`:  
`#>apt-get install nfs-kernel-server`
2. Exporte la carpeta del servidor de imágenes con los mandatos siguientes:  
`#>mkdir /imageserver`  
`#vi /etc/exports`  
`#>service nfs-server restart`

---

## Configuración del almacén de modelos

Es necesario que configure el almacén de modelos en la máquina perimetral. El almacén de modelos es un repositorio para modelos ejecutables distribuidos desde la aplicación central.

### Antes de empezar

Añada la línea siguiente al archivo de exportación:  
`/modelstore "dirección IP"(rw, sync, no_root_squash, no_all_squash)`. Por ejemplo, añada la siguiente línea al archivo:  
`/modelstore 10.173.0.0/29 (rw, sync, no_root_squash, no_all_squash)`.

### Procedimiento

1. Exporte la carpeta del almacén de modelos utilizando los mandatos siguientes:  
`#>mkdir /modelstore`  
`#vi /etc/exports`  
`#>service nfs-server restart`
2. Ejecute los mandatos siguientes:  
`sudo pip install flask`

```
sudo pip install gevent
sudo pip install requests
sudo pip install pyinotify
sudo pip install opencv-python
sudo pip install lmbd
sudo apt-get install dos2unix
```

---

## Registro del sistema perimetral en la aplicación central

Después de configurar la máquina perimetral, es necesario que la registre en la aplicación central. Puede crear un sistema perimetral o editar uno ya existente. Los sistemas perimetrales se utilizan para ejecutar un modelo de puntuación.

### Procedimiento

1. En el Administrador de modelos, seleccione **Mis datos > Sistemas perimetrales**.
2. Seleccione **Crear nuevo sistema perimetral** e indique el nombre del sistema perimetral.
3. Indique la dirección IP, el nombre de usuario de SSH y la contraseña de la máquina perimetral. Es necesario que la aplicación central pueda acceder a la dirección IP. El nombre de usuario de SSH y la contraseña se utilizan para iniciar la sesión en la máquina perimetral y desplegar el controlador del sistema perimetral. El usuario de SSH tiene que ser un usuario root o tener privilegios SUDO.
4. En el recuadro Jerarquía física, añada la jerarquía de celdas a los perfiles. La jerarquía implica que el sistema perimetral incluye dichas celdas y que las imágenes capturadas en ellas se enviarán al sistema perimetral para ser puntuadas. Puede seleccionar una de las combinaciones siguientes: planta, planta y línea, o planta, línea y celda. Si especifica planta, se incluirán todas las celdas que se encuentren bajo la planta. Si elige planta y línea, se incluirán todas las celdas que se encuentren bajo planta y línea. Si un perfil no es suficiente, puede añadir más.
5. Seleccione **Aceptar** para crear el sistema perimetral. Si la dirección IP, el nombre de usuario de SSH o la contraseña no son correctos y aparece un mensaje de error, asegúrese de que los valores sean correctos y de que la máquina perimetral se pueda conectar.
6. Seleccione **Crear**. El controlador perimetral y el motor de puntuación se despliegan en la máquina perimetral y se añade el sistema perimetral a la lista registrada.
7. Lleve a cabo un inicio de sesión SSH para el sistema perimetral.
8. Ejecute el mandato siguiente:  

```
ps aux | grep python
```

El resultado debe incluir el siguiente proceso Python para el controlador perimetral:  

```
python controler.py.
```
9. Si no encuentra el proceso Python en la máquina perimetral, hay un problema al iniciar el controlador perimetral. Consulte los archivos de registro *carpeta\_despliegue/vi\_edge-bin\_vi/vi\_edge/nohup.out*. Intente iniciar el controlador perimetral ejecutando los siguientes mandatos:  

```
cd carpeta_despliegue/vi_edge-bin_vi/vi_edge/
nohup python controler.py &
```
10. Ejecute el siguiente mandato de nuevo para buscar procesos Python para el motor de puntuación:

```
ps aux | grep python
```

El resultado debe incluir los siguientes procesos Python:

```
python
```

```
carpeta_despliegue/vi_edge-bin_vi/vi_score_engine_restful/model  
/run.py 2001 0
```

```
python
```

```
carpeta_despliegue/vi_edge-bin_vi/vi_score_engine_restful/model  
/run.py 2002 0
```

```
python
```

```
carpeta_despliegue/vi_edge-bin_vi/vi_score_engine_restful/route  
/run.py 5005 2001 2002.
```

11. Si no encuentra estos procesos Python en la máquina perimetral, hay un problema al iniciar el controlador perimetral. Consulte los archivos de registro *carpeta\_despliegue*/vi\_edge-bin\_vi/vi\_score\_engine\_restful/route/log.txt y *carpeta\_despliegue*/vi\_edge-bin\_vi/vi\_score\_engine\_restful/model/log.txt. Intente iniciar el motor de puntuación ejecutando el mandato siguiente:

```
carpeta_despliegue/vi_edge-bin_vi/vi_score_engine_restful/startEngine.sh
```

12. Ejecute el siguiente mandato de nuevo para buscar procesos Python para el motor de puntuación de Faster-RCNN y SSD:

```
ps aux | grep python
```

El resultado debe incluir los siguientes procesos Python:

```
python run.py 5060 1 FRCNN
```

```
python run.py 5061 1 SSD
```

13. Si no encuentra estos procesos Python en la máquina perimetral, hay un problema al iniciar el controlador perimetral. Busque los archivos de registro *frcnn\_log.txt* y *ssd\_log.txt* en *carpeta\_despliegue*/vi\_edge-bin\_vi/vi\_obj\_detection/RESTAPI/model. Intente iniciar el motor de puntuación ejecutando el mandato siguiente:

```
cd deployment_folder/vi_edge-bin_vi/vi_obj_detection/RESTAPI/model
```

```
nohup python run.py 5060 1 FRCNN > frcnn_log.txt 2>&1 &
```

```
nohup python run.py 5061 1 SSD > ssd_log.txt 2>&1 &
```

---

## Capítulo 3. Creación y uso de modelos

Los modelos se crean para recopilar información histórica sobre imágenes y defectos. La información se envía a la aplicación central y se utiliza para dar formación al modelo. Tras la formación, el modelo se distribuye a los sistemas perimetrales para su inspección. La información se utiliza para dar formación al modelo. Tras la formación, el modelo debe validarse para poder desplegarse en un sistema perimetral. La validación del modelo proporciona información de precisión del modelo. Puede haber varias versiones de un modelo. Los modelos pueden compartir información de defectos, pero tienen distintos archivos de imagen de distintas líneas de producto. Puede repetir la formación de un modelo para intentar obtener una mayor precisión del modelo, para que la versión del modelo pueda sustituirse por una versión del modelo más reciente. Hay dos tipos de implementaciones de modelos: el modelo de clasificación y el modelo de detección de objetos. El modelo de detección de objetos no permite la repetición de la formación.

---

### Estructura de archivos de imagen comprimidos

Para poder añadir imágenes históricas de grupos de imágenes, debe tener archivos que contengan los archivos de imagen que necesita para el modelo de clasificación o el modelo de detección de objetos.

#### Modelo de clasificación

Añada las imágenes a los archivos comprimidos. Un archivo comprimido debe contener todas las imágenes que pertenecen al mismo grupo de imágenes. Debe colocar todas las imágenes en una estructura plana sin subcarpetas en el archivo comprimido.

#### Modelo de detección de objetos

La estructura del modelo de detección de objetos debe contener dos carpetas comprimidas en un archivo comprimido. Una carpeta debe denominarse JPEGImages y la otra debe denominarse Annotations. Además de las dos carpetas, el archivo comprimido también debe contener un archivo labels.txt.

Añada todos los archivos de imagen a la carpeta JPEGImages. Añada todos los archivos de anotaciones a la carpeta Annotations. Un archivo de anotaciones debe tener el mismo nombre que su archivo de imagen. Los archivos debe tener un formato XML. La siguiente información es un ejemplo de un archivo de anotaciones:

```
<annotation>
  <folder>JPEGImages</folder>
  <filename>000001.jpg</filename>
  <source>
    <database>Unknown</database>
  </source>
  <size>
    <width>864</width>
    <height>1296</height>
    <depth>3</depth>
  </size>
  <segmented>0</segmented>
</object>
```

```

<name>defect1</name>
<pose>Unspecified</pose>
<truncated>0</truncated>
<difficult>0</difficult>
<bndbox>
  <xmin>474</xmin>
  <ymin>368</ymin>
  <xmax>540</xmax>
  <ymax>448</ymax>
</bndbox>
</object>
<object>
  <name>defect2</name>
  <pose>Unspecified</pose>
  <truncated>0</truncated>
  <difficult>0</difficult>
  <bndbox>
    <xmin>303</xmin>
    <ymin>387</ymin>
    <xmax>369</xmax>
    <ymax>452</ymax>
  </bndbox>
</object>
</annotation>

```

El archivo `labels.txt` contiene los nombres de todos los tipos de defecto que hay en la carpeta de imágenes comprimida. Cada defecto debe estar en una línea distinta, como se muestra en el siguiente ejemplo:

```

defect1
defect2
defect3

```

---

## Adición de imágenes históricas para grupos de imágenes

El administrador de modelos puede crear grupos de imágenes utilizando imágenes históricas. para dar formación a un modelo.

### Procedimiento

1. Seleccione **Datos > Grupos de imágenes > Nuevo grupo de imágenes**.
2. Añada un nombre de grupo de imágenes exclusivo y una descripción, seleccione el tipo de grupo de imágenes y seleccione **Siguiente**.

Como tipo de grupo de imágenes, Características individuales significa que todas las imágenes que hay en el grupo tienen el mismo tipo de defecto. Para el tipo de grupo Característica individuales, cada imagen en el grupo tiene un defecto. Varias características significa que todas las imágenes que hay en el grupo tienen más de un defecto. Para el tipo de grupo Varias características, los tipos de defecto pueden ser iguales o diferentes. Como tipo de grupo de imágenes, Ningún defecto significa que las imágenes en el grupo no tienen defectos.

**Nota:** Una vez cargado el conjunto de imágenes, el tipo de grupo de imágenes que se selecciona no se puede cambiar.

3. En el panel Conjuntos de imágenes, añada imágenes y seleccione **Añadir grupo de imágenes**.
4. En el panel Grupos de imágenes, abra al defecto que ha añadido y seleccione **Vista previa** para ver los archivos de imágenes.

## Resultados

En el panel Grupos de imágenes, puede seleccionar un grupo de imágenes y seleccione Editar. Puede añadir o suprimir los conjuntos de imágenes, actualizar el nombre del grupo de imágenes o la descripción, o cambiar el tipo de grupo de imágenes.

**Nota:** Si el conjunto de imágenes se cita en una instancia de modelo, el conjunto de imágenes no puede suprimirse.

---

## Creación de solicitudes de modelos

Una vez añadidos los tipos de defectos, el administrador de modelos crea una solicitud de modelo. La solicitud de modelo se envía a un experto en datos, que recibe el modelo. El experto en datos da formación al modelo.

### Acerca de esta tarea

Puede ver los detalles de un modelo en el separador **Todos los modelos**. Los detalles del modelo incluyen las versiones del modelo. Cada conjunto de imágenes diferente crea distintas versiones del modelo.

El nombre del modelo y el tipo de producto son exclusivos. No se pueden utilizar nombres de modelo existentes ni tipos de productos al crear un nuevo modelo. Sin embargo, el mismo nombre de modelo puede existir en dos listas independientes para representar una versión diferente del mismo modelo. Cuando el nuevo modelo se distribuye a los sistemas perimetrales, se sustituye la versión antigua.

### Procedimiento

1. Seleccione **Crear nuevo modelo**.
2. En el separador **General**, actualice la información. El tipo de producto se utiliza para correlacionar el modelo. El controlador del sistema perimetral selecciona el modelo en función de la información del tipo de producto que aparece en la imagen.
3. En el separador **Políticas globales**, establezca la política de repetición de formación y los valores de política de inspección manual. Si selecciona Repetir formación automáticamente, la repetición de la formación se inicia automáticamente cuando se cumplen las condiciones. Si selecciona Repetir formación manualmente, puede seleccionar el nivel de confianza del administrador de modelos para repetir la formación manualmente.
4. En el separador **Añadir defectos**, seleccione los conjuntos de imágenes que desee utilizar para dar formación al modelo y seleccione **Hecho**. El estado de la versión del modelo se establece en Borrador.
5. En el panel Todos los modelos, vea el modelo. Si el estado de la versión del modelo se establece en Borrador, puede seleccionar un grupo de imágenes y editarlo.
6. Seleccione **Solicitar formación**. El estado de la versión del modelo cambia de Borrador para Enviado.

---

## Modelos formados

El experto en datos da formación al modelo utilizando la descripción del modelo y los conjuntos de imágenes. El administrador de modelos valida el modelo y acepta o rechaza el modelo formado. El administrador de modelos valida el modelo utilizando conjuntos de imágenes de validación.

Es aconsejable que el experto en datos utilice la herramienta de desarrollo de NVIDIA para dar formación al modelo. El experto en datos empaqueta el resultado del modelo de Caffe y los archivos de parámetros para, a continuación, enviar el archivo comprimido al administrador de modelos. El administrador de modelos selecciona el modelo y adjunta el archivo comprimido. Una vez cargado el archivo comprimido, el estado del modelo cambia a Formado.

## Estructura de los archivos de modelo

Visual Insights da soporte a la clasificación CNN (Convolutional Neural Network) y a los tipos de modelo de detección de objetos.

### Modelos de clasificación CNN

El modelo de clasificación CNN debe tener un único archivo comprimido y contener la estructura de directorios y los archivos correctos.

### El archivo de modelo comprimido

El archivo de modelo comprimido debe contener los siguientes directorios y archivos.

- `model.config` (archivo, necesario)
- `sink.config` (archivo, necesario)
- `parameter.config` (archivo, opcional)
- `cnet` (directorio). El directorio `cnet` debe contener los siguientes archivos:
  - `labels.txt` (archivo, necesario)
  - `deploy.prototxt` (archivo, necesario)
  - `mean.binaryproto` (archivo, necesario)
  - `info.json` (archivo, necesario)
  - `snapshot.caffemodel` (archivo, necesario)
  - `solver.prototxt` (archivo, necesario)
  - `train_val.prototxt` (archivo, necesario)

Cada archivo debe tener la estructura y las palabras clave correctas. Los archivos se describen en las siguientes secciones.

### `model.config`

El texto siguiente es un ejemplo del contenido del archivo `model.config`. Las palabras clave aparecen en negrita.

```
submodel{
  module {
    type:"ChipROIExtractor"
    ref_file:"parameter.config"
  }
  module {
    type:"ClassificationNet"
    net_name:"cnet"
  }
}
```

```

}
sink{
  type:"SinkFilter"
  ref_file:"sink.config"
}

```

Este archivo debe tener al menos un módulo que tenga el tipo **ClassificationNet** en el submodelo. La palabra clave `ref_file` apunta a los otros archivos de configuración en el archivo comprimido del modelo de clasificación CNN. La palabra clave `net_name` hace referencia al nombre de carpeta que contiene el modelo CNN. No necesita cambiar la información de `sink` a menos que tenga un nombre diferente para el archivo `sink.config`.

### sink.config

El contenido del archivo `sink.config` es el siguiente. No es necesario editar el contenido.

```

keyword:"position"
keyword:"probableTypes"

```

### labels.txt

El archivo `labels.txt` contiene los nombres de clase con los que se clasifica este modelo de clasificación. Cada nombre de clase debe estar en una línea distinta, como se muestra en el siguiente ejemplo.

```

defect1
defect2
defect3

```

### info.json

El archivo `info.json` contiene toda la información de metadatos del modelo CNN. Todos los nombres de archivo deben coincidir con los nombres de archivo que se incluyen en este modelo. Las palabras clave en el archivo de ejemplo siguiente se muestran en negrita.

```

{
  "caffe flavor": "BVLC",
  "caffe version": "1.0.0-rc5",
  "creation time": "2017-05-02 16:26:18.957631",
  "dataset_id": "20170502-162536-2027",
  "deploy file": "deploy.prototxt",
  "digits version": "4.1-dev",
  "framework": "caffe",
  "id": "20170502-162618-b701",
  "image dimensions": [
    227,
    227,
    3
  ],
  "image resize mode": "squash",
  "job id": "20170502-162618-b701",
  "labels file": "labels.txt",
  "mean file": "mean.binaryproto",
  "model file": "original.prototxt",
  "name": "C_3_227_227",
  "snapshot file": "snapshot_iter_138.caffemodel",
  "solver file": "solver.prototxt",
  "status": "Aborted",
  "train_val file": "train_val.prototxt",
  "username": "coco"
}

```

## Otros archivos

Todos los archivos *nombre\_archivo.prototxt* son archivos de definición de modelo que son necesarios para dar formación a un modelo CNN.

Los archivos *snapshot.caffemodel* y *mean.binaryproto* son archivos de salida que se crean cuando finaliza la formación de modelos.

## Modelo de detección de objetos

El modelo de detección de objetos debe tener un único archivo comprimido y contener la estructura de directorios y los archivos correctos. Están soportados los siguientes algoritmos de detección de objetos: You Only Look Once (YOLO) V2, Faster Region-based Convolutional Neural Network (Faster R-CNN) y Single Shot Multibox Detector (SSD).

### YOLO V2:

El modelo de detección de objetos YOLO V2 debe tener un único archivo comprimido y contener la estructura de directorios y los archivos correctos.

### Archivo de modelo comprimido

El archivo de modelo comprimido debe contener los siguientes archivos:

- `labels.txt`
- `model.config`
- `yolo_final.weights`
- `Yolo.cfg`

Cada archivo debe tener la estructura y las palabras clave correctas. Los archivos se describen en las siguientes secciones.

### labels.txt

El archivo `labels.txt` contiene los nombres de todos los objetos que detecta este modelo de detección de objetos. Cada objeto debe estar en una línea distinta, como se muestra en el siguiente ejemplo.

```
defect1
defect2
defect3
```

### model.config

Las palabras clave en el ejemplo siguiente se muestran en negrita:

```
{
  "modelType": "YOLO",
  "modelCfg": "Yolo.cfg",
  "model": "yolo_final.weights",
  "parameters": {
    "iteration": "40000",
    "batchSize": 16,
    "learningRate": 0.001,
    "subBatchSize": 2,
    "steps": "100, 15000, 25000,35000",
    "scales": "-1,10, 0.1, 0.1"
  }
}
```

El valor de `modelType` siempre es YOLO. El valor de `modelCfg` es el nombre del archivo de definición de red de aprendizaje profundo. El valor del modelo es el nombre del archivo de modelo real. Los valores en los parámetros son hiperparámetros del modelo YOLO V2.

### **yolo\_final.weights**

Este archivo contiene la salida después de formar el modelo YOLO. El nombre de archivo debe coincidir con la definición de **model** en `model.config`.

### **Yolo.cfg**

Este archivo de definición de modelo incluye definiciones de red, hiperparámetros y valores de ancla. Puede encontrar una plantilla de este archivo en el directorio de instalación de YOLO `darknet/cfg/`. Este archivo debe coincidir con el archivo `weights`.

### **Faster R-CNN:**

El modelo de detección de objetos Faster R-CNN debe tener un único archivo comprimido y contener la estructura de directorios y los archivos correctos.

### **Archivo de modelo comprimido**

El archivo de modelo comprimido debe contener los siguientes archivos:

- `labels.txt`
- `faster_rcnn_final.caffemodel`
- `model.config`
- `stage1_fast_rcnn_solver30k40k.pt`
- `stage1_fast_rcnn_train.pt`
- `stage1_rpn_solver60k80k.pt`
- `stage1_rpn_train.pt`
- `stage2_fast_rcnn_solver30k40k.pt`
- `stage2_fast_rcnn_train.pt`
- `stage2_rpn_solver60k80k.pt`
- `stage2_rpn_train.pt`
- `faster_rcnn_test.pt`

Cada archivo debe tener la estructura y las palabras clave correctas. Los archivos se describen en las siguientes secciones.

### **labels.txt**

El archivo `labels.txt` contiene los nombres de todos los objetos que detecta este modelo de detección de objetos. Cada objeto debe estar en una línea distinta, como se muestra en el siguiente ejemplo.

```
defect1  
defect2  
defect3
```

## **faster\_rcnn\_final.caffemodel**

Este archivo contiene la salida después de formar el modelo Faster R-CNN. El nombre de archivo debe coincidir con la definición de **model** en `model.config`.

## **model.config**

Las palabras clave en el ejemplo siguiente se muestran en negrita:

```
{
  "modelType": "FRCNN",
  "model": "faster_rcnn_final.caffemodel",
  "solvers": "stage1_rpn_solver60k80k.pt,stage1_fast_rcnn_solver30k40k.pt,
stage2_rpn_solver60k80k.pt,stage2_fast_rcnn_solver30k40k.pt",
  "net_file": "stage1_rpn_train.pt,stage1_fast_rcnn_train.pt,
stage2_rpn_train.pt,stage2_fast_rcnn_train.pt",
  "deploy_net": "faster_rcnn_test.pt",
  "parameters": {
    "iteration": "40000,80000,40000,80000",
    "learningRate": 0.001,
    "stepsize": "10000",
    "gamma": "0.1"
  }
}
```

El valor de `modelType` siempre es `FRCNN`. El valor del modelo es el nombre del archivo de modelo. El valor de `solvers` es la lista de archivos solver que se utilizan durante la formación del modelo. El valor de `net_file` es la lista de archivos de definición de red. El valor de `deploy_net` es el nombre de la definición de red de puntuación. Los valores en los parámetros son todos hiperparámetros del modelo Faster R-CNN.

## **Archivos \*.pt**

Los archivos con la extensión `.pt` son archivos de definición de modelo. El modelo `pascal_voc` que proporciona Faster R-CNN está soportado. El archivo de plantilla puede encontrarse en el directorio de instalación de Faster R-CNN `models/pascal_voc/nombre_red/faster_rcnn_alt_opt/`, donde *nombre\_red* es `ZF` o `VGG16`.

## **SSD:**

El modelo de detección de objetos SSD (Single Shot MultiBox Detector) debe tener un único archivo comprimido y contener la estructura de directorios y los archivos correctos.

## **Archivo de modelo comprimido**

El archivo de modelo comprimido debe contener los siguientes archivos:

- `labels.txt`
- `solver.prototxt`
- `deploy.prototxt`
- `model.config`
- `SSD.caffemodel`

Cada archivo debe tener la estructura y las palabras clave correctas. Los archivos se describen en las siguientes secciones.

### labels.txt

El archivo labels.txt contiene los nombres de todos los objetos que detecta este modelo de detección de objetos. Cada objeto debe estar en una línea distinta, como se muestra en el siguiente ejemplo.

```
defect1  
defect2  
defect3
```

### solver.prototxt

Este archivo contiene todos los hiperparámetros para el modelo SSD.

### deploy.prototxt

Este archivo contiene la definición de red del modelo formado.

### model.config

Las palabras clave en el ejemplo siguiente se muestran en negrita:

```
{  
  "modelType": "SSD",  
  "seldef_parameters": {  
    "TestRatio": 0.3,  
    "Train_batch_size": 8,  
    "Validation_batch_size": 8  
  }  
}
```

El valor de modelType siempre es SSD. Los valores de seldef\_parameters son los parámetros que se utilizan para la formación y la validación.

### SSD.caffemodel

Este archivo contiene la salida después de formar el modelo SSD.

---

## Modelos validados

El administrador de modelos valida la versión de modelo y acepta o rechaza el modelo formado. El administrador de modelos valida el modelo utilizando conjuntos de imágenes de validación. Una vez validado un modelo, se puede utilizar y desplegar.

Puede validar una versión de modelo que tenga un estado Formado. Una versión de modelo formada o para la que se ha repetido la formación puede desencadenar el proceso de validación. Cuando valida un informe, debe gestionar los conjuntos de imágenes. Cada grupo de imágenes debe tener al menos un conjunto de imágenes de validación para poder validar el modelo. Debe utilizar conjuntos de imágenes diferentes y conjuntos de imágenes de formación para poder validar la versión de modelo. Para empezar el proceso de validación, seleccione **Validar**.

Una vez finalizado el proceso de validación, se genera un informe que muestra la precisión del modelo. Puede crear los tipos de informes siguientes:

### Informe de modelo de clasificación

En el informe de modelo de clasificación, una imagen tiene un defecto como máximo. La matriz de confusión se utiliza para generar un informe donde cada columna representa un tipo de grupo de imágenes real en los

conjuntos de datos de validación. Cada fila representa el tipo de grupo de imágenes previsto. La última fila del gráfico representa los resultados de agregación.

### **Informe de modelo de detección de objetos**

En el informe de modelo de detección de objetos, una imagen tiene varios defectos. En este informe, se calculan la precisión media y la rellamada para indicar la precisión del modelo.

Para ver el informe, seleccione **Mostrar informe**. En la ventana Informe de validación, puede revalidar, rechazar o aceptar y desplegar el informe.

Si se rechaza la primera versión de modelo, puede adjuntar un nuevo archivo de modelo. Si se rechaza una versión de modelo que no es la primera versión, el administrador de modelos puede repetir la formación de una nueva versión de modelo.

---

## **Distribución de modelos formados en sistemas perimetrales**

Una vez que el administrador de modelos haya aceptado la versión del modelo formado, este se distribuye a los sistemas perimetrales para su inspección.

### **Procedimiento**

1. Seleccione una versión de modelo con un estado Aceptado. Seleccione **Desplegar**.
2. Seleccione el filtro de planta, línea o celda para buscar los sistemas perimetrales donde se desplegará el modelo.
3. Seleccione los sistemas perimetrales desde la Lista de candidatos y muévalos a la Lista de seleccionados.
4. Seleccione **Desplegar**.

---

## **Modelos de repetición de formación**

Cuando repite la formación de un modelo, se crea una nueva versión del modelo. Cuando crea una solicitud de modelo, define la política de repetición de formación. La política de repetición de formación es la condición que desencadena la repetición de formación automática. Si hay algún problema sobre la precisión del modelo actual, el administrador de modelos puede seleccionar manualmente los archivos de imagen para desencadenar el proceso de repetición de formación.

### **Procedimiento**

1. Seleccione una versión de modelo que tenga un estado Desplegado y seleccione **Repetir formación**.
2. Gestione los conjuntos de imágenes. Cada grupo de imágenes debe tener al menos un conjunto de imágenes de repetición de formación para poder repetir la formación del modelo.

**Nota:** Debe utilizar conjuntos de imágenes diferentes para poder repetir la formación de la versión de modelo.

3. Seleccione **Repetir formación** para comenzar el proceso de repetición de formación.

## **Qué hacer a continuación**

El administrador de modelos valida si el modelo para el que se ha repetido la formación se ha aceptado. El administrador de modelos puede desplegar la nueva versión y la versión de modelo anterior ya no se despliega.



---

## Capítulo 4. Comprobación de los resultados de la inspección

Después de enviar los resultados de la inspección a la aplicación central, el inspector y el supervisor de inspectores pueden ir al separador Comprobación de defectos para ver y filtrar los resultados de la inspección y realizar los cambios pertinentes.

---

### Imágenes

El inspector y el supervisor de inspectores pueden ver las imágenes para comprobar si están clasificadas como defectos existentes o no, y para averiguar si alguien más las ha revisado. La visualización de las imágenes determina lo que deben hacer el inspector o el supervisor cuando comprueban los defectos.

El inspector ve las imágenes que se han comprobado y las que no. Que una imagen no esté comprobada quiere decir que el modelo ha señalado la imagen, pero esta no ha sido comprobada por un inspector. Las imágenes comprobadas han sido señaladas por el modelo y comprobadas por un inspector.

El inspector ve las imágenes que se han confirmado y las que no. Que una imagen no esté confirmada quiere decir que el modelo ha señalado la imagen, pero esta no ha sido confirmada por un inspector. Las imágenes confirmadas han sido señaladas por el modelo y confirmadas por un inspector.

El supervisor de inspectores puede ver los objetos y los objetos desconocidos. Los objetos desconocidos implican que la imagen ha sido marcada por un inspector como defecto desconocido, ya que el inspector no ha clasificado la imagen como defecto existente. Estas imágenes aparecen resaltadas en la lista del supervisor de inspectores.

---

### Filtrado de defectos

El inspector y el supervisor de inspectores pueden aplicar filtros a la visión general de la celda y la lista de defectos.

#### Procedimiento

1. En la ventana Todas las estaciones de trabajo, seleccione una estación de trabajo para ver la lista de objetos no confirmados, confirmado y desconocidos.
2. Seleccione el icono de filtro.
3. Escriba un valor para una condición para establecer el filtro y, a continuación, seleccione el icono de añadir. El filtro se aplica a la lista inmediatamente.

---

### Comprobación de defectos

El inspector y el supervisor de inspectores revisan los resultados de la inspección y aplican los cambios necesarios.

#### Acerca de esta tarea

Al seleccionar una imagen, se visualizan el defecto candidato y su confianza correspondiente. Aparece seleccionado el primer defecto de forma predeterminada.

El inspector puede seleccionar "desconocido" como tipo de defecto si no conoce el tipo de defecto.

### **Procedimiento**

1. Seleccione una imagen para ver los detalles de la imagen y los resultados de la inspección.
2. Seleccione **Editar zoom** para acercar y alejar la imagen o arrastrar la imagen para localizar una posición.
3. Seleccione **Establecer zoom** para volver al modo de edición. Puede añadir, cambiar el tamaño, mover y ver los detalles de un recuadro de defecto.
4. Seleccione el recuadro de defecto que desea confirmar y vea los detalles del tipo de defecto y el nivel de confianza. Puede cambiar el tipo de defecto o suprimir la posición.
5. Seleccione **Confirmar**.
6. Si la imagen no pertenece a ningún defecto existente, el supervisor de inspectores puede crear un nuevo grupo de imágenes. El nuevo defecto se añade a la lista de candidatos para aquellas imágenes que se encuentren en el mismo modelo.

---

## Capítulo 5. Panel de control del indicador clave de rendimiento

Un supervisor de inspectores utiliza el panel de control del indicador clave de rendimiento (KPI) para ayudar a gestionar a todos los inspectores. El supervisor de inspectores también utiliza el panel de control del KPI para comprobar tasa de defectos a nivel de imagen y la tasa de defectos a nivel de ubicación. Las métricas proporcionan información que permiten al usuario pedir al equipo de TI que repitan la formación del modelo o ajusten el procedimiento de fabricación.

El panel de control del KPI se encuentra en el separador KPI. Puede seleccionar todas las estaciones de trabajo o una estación de trabajo específica. Esta selección tiene repercusión en el ámbito en el que está trabajando. También puede conmutar entre las vistas en tiempo real e históricas. En la vista en tiempo real, los datos de KPI se renuevan cada 5 segundos. Los valores de KPI incluyen los defectos por unidad y la tasa de defectos. Los defectos por unidad se calculan como un determinado número de defectos dividido por el número de imágenes total. El valor de defectos por unidad representan tasa de apariciones de un tipo de defecto. La tasa de defectos se calcula como el número de imágenes con uno o varios defectos dividido por el número de imágenes total. La tasa de defectos representa la tasa de defectos del producto. Cada línea del gráfico representa el valor de KPI en el intervalo de 5 segundos actual. La vista histórica muestra los datos de KPI históricos. Puede editar la fecha de inicio y la fecha de finalización para determinar el rango de tiempo. Los KPI en la vista histórica incluyen los defectos por unidad y la tasa de defectos.

La vista histórica tiene tres granularidades: por hora, a diario y mensual. En el gráfico por horas, cada punto representa 1 hora. Por ejemplo, un valor de KPI de 24 puntos equivale a 24 horas. En la gráfica diaria, cada punto representa un día. Por ejemplo, un valor de KPI de 30 puntos equivale a 30 días. En la gráfica mensual, cada punto representa un mes. Por ejemplo, un valor de KPI de 12 puntos equivale a 12 meses. Hay tres relaciones entre el rango de tiempo seleccionado y la granularidad. Si hay demasiados puntos en una granularidad para un rango de tiempo seleccionado, esa granularidad está inhabilitada hasta que acorta el rango de tiempo. Si desea renovar el gráfico, puede cambiar el rango de tiempo o pulsar **Renovar**. Periódicamente, el servidor calcula los datos de KPI, por lo tanto, se produce un retardo en función de la configuración. El periodo predeterminado para calcular el KPI histórico es de 1 hora.

De forma predeterminada, sólo se visualizan los cinco primeros tipos de defectos en el gráfico Defectos por unidad. Si desea comprobar otros tipos de defectos, puede seleccionar uno o más tipos de defectos en el gráfico y, a continuación, pulsar **Mostrar tendencia**. Un nuevo gráfico de KPI muestra los tipos de defectos seleccionados. Si el valor de KPI de un tipo de defecto es 0, no se puede seleccionar para mostrar una tendencia.



---

## Avisos

Esta información se ha desarrollado para productos y servicios que se ofrecen en los Estados Unidos. Este material puede estar disponible por parte de IBM en otros idiomas. Sin embargo, es posible que tenga obligación de tener una copia del producto o de la versión del producto en dicho idioma para acceder a él.

Puede que IBM no ofrezca en algunos países los productos, servicios o características que se explican en este documento. Póngase en contacto con el representante local de IBM para obtener información sobre los productos y servicios disponibles actualmente en su área. Cualquier referencia hecha a un producto, programa o servicio de IBM no implica ni pretende indicar que sólo pueda utilizarse ese producto, programa o servicio de IBM. En su lugar, se puede utilizar cualquier producto, programa o servicio funcionalmente equivalente que no infrinja ninguno de los derechos de propiedad intelectual de IBM. Sin embargo, será responsabilidad del usuario evaluar y verificar el funcionamiento de cualquier programa, producto o servicio que no sea de IBM.

Puede que IBM tenga patentes o solicitudes de patentes pendientes relacionadas con los temas principales que se tratan en este documento. La posesión de este documento no le otorga ninguna licencia sobre estas patentes. Puede enviar sus consultas sobre licencias, por escrito, a:

*IBM Director of Licensing  
IBM Corporation  
North Castle Drive, MD-NC119  
Armonk, NY 10504-1785  
EE.UU.*

Para consultas sobre licencias en las que se solicite información sobre el juego de caracteres de doble byte (DBCS), póngase en contacto con el departamento de propiedad intelectual de IBM de su país o envíe las consultas, por escrito, a:

*Intellectual Property Licensing  
Legal and Intellectual Property Law  
IBM Japan Ltd.  
19-21, Nihonbashi-Hakozakicho, Chuo-ku  
Tokio 103-8510, Japón*

INTERNATIONAL BUSINESS MACHINES CORPORATION PROPORCIONA ESTA PUBLICACIÓN "TAL CUAL" SIN GARANTÍA DE NINGÚN TIPO, NI EXPLÍCITAS NI IMPLÍCITAS, INCLUIDAS, PERO SIN LIMITARSE A, LAS GARANTÍAS IMPLÍCITAS DE NO INFRACCIÓN, COMERCIALIZACIÓN O IDONEIDAD PARA UN PROPÓSITO CONCRETO. Algunas legislaciones no contemplan la exclusión de garantías, ni implícitas ni explícitas, por lo que puede haber usuarios a los que no afecte esta declaración.

Esta información podría incluir imprecisiones técnicas o errores tipográficos. Periódicamente se realizan cambios en la información que aquí se presenta; tales cambios se incorporarán en nuevas ediciones de la publicación. IBM podría realizar mejoras y/o cambios en el producto o productos y/o programa o programas que se describen en esta publicación en cualquier momento y sin previo aviso.

Cualquier referencia en este documento a sitios web que no son de IBM se proporciona únicamente para su comodidad y no significa en modo alguno que se recomiende visitar dichos sitios web. Los materiales de dichos sitios web no forman parte de los materiales para este producto de IBM y el uso de estos sitios web solo es responsabilidad del cliente.

IBM puede utilizar o distribuir la información que el cliente le suministre del modo que considere conveniente, sin incurrir por ello en ninguna obligación para con el cliente.

Los titulares de licencias de este programa que deseen obtener información acerca de este con el fin de permitir: (i) el intercambio de información entre programas creados independientemente y otros programas (incluido el presente) y (ii) la utilización mutua de la información que se ha intercambiado, deben ponerse en contacto con:

*IBM Director of Licensing  
IBM Corporation  
North Castle Drive, MD-NC119  
Armonk, NY 10504-1785  
EE.UU.*

Esta información podría estar disponible, de acuerdo con los términos y condiciones aplicables, incluyendo en algunos casos el pago de una tarifa.

El programa bajo licencia que se describe en este documento y todo el material bajo licencia disponible para éste los proporciona IBM en función de lo dispuesto en el IBM Customer Agreement, el Acuerdo Internacional de Licencia de Programa de IBM o cualquier otro acuerdo equivalente celebrado entre ambas partes.

Los datos de rendimiento y los ejemplos de clientes citados se presentan solamente a efectos ilustrativos. Los resultados reales de rendimiento podrían variar en función de las configuraciones específicas y las condiciones de funcionamiento.

La información relativa a los productos que no son de IBM ha sido obtenida de los proveedores de dichos productos, sus anuncios publicados u otras fuentes públicas disponibles. IBM no ha probado estos productos y no puede confirmar la precisión de rendimiento, compatibilidad ni otras afirmaciones relacionadas con productos que no son de IBM. Las preguntas sobre las prestaciones de los productos que no son de IBM deben dirigirse a los proveedores de dichos productos.

Esta documentación se suministra solo a efectos de planificación. La información aquí contenida puede cambiar antes de que se comercialicen los productos descritos.

Esta información contiene ejemplos de datos e informes que se utilizan en operaciones empresariales de uso habitual. Para ilustrarlas de la forma más completa posible, los ejemplos incluyen nombres de personas, compañías, marcas y productos. Todos estos nombres son ficticios y cualquier parecido con personas o empresas comerciales reales es mera coincidencia.

#### LICENCIA DE COPYRIGHT:

Esta información contiene programas de aplicaciones de ejemplo en código fuente, que ilustran técnicas de programación en las distintas plataformas operativas. Puede copiar, modificar y distribuir estos programas de muestra como considere

oportuno, sin tener que pagar a IBM, con intención de desarrollar, utilizar, comercializar o distribuir programas de aplicación de acuerdo con la interfaz de programación de aplicaciones para la plataforma operativa para la cual están escritos los programas de muestra. Los ejemplos no se han probado minuciosamente bajo todas las condiciones. Por lo tanto, IBM no puede garantizar ni dar por sentada la fiabilidad, la capacidad de servicio ni el funcionamiento de dichos programas. Los programas de ejemplo se ofrecen "TAL CUAL", sin garantía de ningún tipo. IBM no será responsable de los daños derivados del uso de los programas de muestra.

---

## Marcas registradas

IBM, el logotipo de IBM e [ibm.com](http://ibm.com) son marcas registradas de International Business Machines Corp., registradas en muchas jurisdicciones en todo el mundo. Otros nombres de productos y servicios pueden ser marcas registradas de IBM o de otras compañías. Encontrará una lista actualizada de marcas registradas de IBM en el apartado "Información de copyright y marcas registradas" de la web [www.ibm.com/legal/copytrade.shtml](http://www.ibm.com/legal/copytrade.shtml).

Java y todas las marcas registradas y logotipos basados en Java son marcas registradas de Oracle o sus filiales.

Linux es una marca registrada de Linus Torvalds en los Estados Unidos o en otros países.

Microsoft, Windows, Windows NT y el logotipo de Windows son marcas registradas de Microsoft Corporation en EE.UU. y/o en otros países.

UNIX es una marca registrada de The Open Group en EE.UU. y en otros países.

---

## Términos y condiciones de la documentación de producto

Se otorga permiso para el uso de estas publicaciones si se cumplen estos términos y condiciones.

### Aplicabilidad

Estos términos y condiciones completan los términos y condiciones de uso del sitio web de IBM.

### Uso personal

Puede reproducir estas publicaciones para su uso personal, no comercial siempre que se conserven todos los avisos de propiedad. No puede distribuir, visualizar o realizar trabajo derivado de estas publicaciones, ni de ninguna parte de las mismas, sin el expreso consentimiento de IBM.

### Uso comercial

Puede reproducir, distribuir y mostrar estas publicaciones únicamente dentro de su empresa siempre que se preserven todos los avisos de la marca registrada. No está autorizado a realizar acciones derivadas de estas publicaciones, ni a reproducir, distribuir ni mostrar estas publicaciones o fragmentos de las mismas fuera de la empresa, sin el consentimiento expreso de IBM.

## Derechos

Excepto de la forma expresamente otorgada en este permiso, no se otorga ningún otro permiso, licencia o derecho, ni explícito ni implícito, a las publicaciones o cualquier información, datos, software u otra propiedad intelectual aquí contenida.

IBM se reserva el derecho de retirar los permisos otorgados aquí siempre que, a su discreción, la utilización de las publicaciones se considere perjudicial para sus intereses o, según determine IBM, las instrucciones anteriores no se sigan correctamente.

Queda prohibido descargar, exportar o reexportar esta información si no se cumplen íntegramente todas las leyes y regulaciones aplicables, incluidas las leyes y regulaciones de exportación de los Estados Unidos.

IBM NO PROPORCIONA NINGÚN TIPO DE GARANTÍA SOBRE EL CONTENIDO DE ESTAS PUBLICACIONES. ESTAS PUBLICACIONES SE PROPORCIONAN "TAL CUAL" SIN NINGÚN TIPO DE GARANTÍA, IMPLÍCITA NI EXPLÍCITA, INCLUIDAS, PERO SIN LIMITARSE A, LAS GARANTÍAS IMPLÍCITAS DE COMERCIALIZACIÓN, IDONEIDAD PARA UN PROPÓSITO CONCRETO O NO INFRACCIÓN.

---

## Declaración de privacidad en línea de IBM

Los productos de IBM Software, incluidas las soluciones de software como servicio ("ofertas de software"), pueden utilizar cookies u otras tecnologías para recopilar información de uso del producto, ayudar a mejorar la experiencia del usuario final, adaptar las interacciones con el usuario final u otros fines. En muchos casos, las ofertas de software no recopilan información de identificación personal. Algunas de nuestras ofertas de software pueden ayudarle a recopilar información de identificación personal. Si esta oferta de software utiliza cookies para recopilar información de identificación personal, más adelante se incluirá información específica sobre el uso de cookies por parte de esta oferta.

En función de la configuración desplegada, esta oferta de software puede utilizar cookies de sesión y persistentes que recopilan los nombres, nombres de usuario, contraseñas u otra información de identificación personal de todos los usuarios con fines de gestión de sesiones, autenticación, configuración de inicio de sesión único u otros fines de seguimiento de uso o funcionales. Estas cookies se pueden inhabilitar, pero, de hacerlo, es posible que se elimine también la funcionalidad que habilitan.

Si las configuraciones desplegadas para esta oferta de software proporcionan al cliente la capacidad de recopilar información de identificación personal de los usuarios finales mediante cookies y otras tecnologías, aquel debería obtener su propio asesoramiento legal acerca de las leyes aplicables a dicha recopilación de datos, incluidos los requisitos de aviso y consentimiento.

Para obtener más información acerca del uso de las distintas tecnologías, incluidas las cookies, para estos fines, consulte la política de privacidad de IBM en <http://www.ibm.com/privacy> y la declaración de política de privacidad de IBM en <http://www.ibm.com/privacy/details>, en la sección llamada "Cookies, Web Beacons and Other Technologies", y la publicación "IBM Software Products and Software-as-a-Service Privacy Statement" en <http://www.ibm.com/software/info/product-privacy>.





Impreso en España