The difference is that Cygwin knows how to process so-called "Win32" pathnames. Loosely speaking, it knows that "\" is a pathname separator and not a shell escape character like it is in Bash. The error you showed is Bash interpreting "\" as an escape character. (Edit: you might try replacing one \ with two, thus escaping the backslash, so that Bash passes the command through to reg.exe correctly.)

Having said that, while Cygwin groks Windows pathnames, it doesn't like to do so. The [documentation](https://cygwin.com/cygwin-ug-net/using.html#pathnames-win32) warns you off from using them. Of course, a little back-sliding now and then doesn't hurt. But while you can run Windows programs in Cygwin, hygiene suggests you should run Windows programs in the CMD.EXE processor and UNIX ones in Cygwin for your long-term sanity.

You can use [cygpath](https://cygwin.com/cygwin-ug-net/cygpath.html) to convert between windows and unix path formats. – [DavidPostill♦](https://superuser.com/users/337631/davidpostill) [Mar 24 '17 at 14:40](https://superuser.com/questions/1191805/can-git-bash-and-cygwin-shell-do-the-same-things#comment1730808_1191820)

* Thanks. The pathname in my commands are used as keys in the registry table, so can "\" in them be replaced with "/"? – [Tim](https://superuser.com/users/9265/tim) [Mar 24 '17 at 19:05](https://superuser.com/questions/1191805/can-git-bash-and-cygwin-shell-do-the-same-things#comment1731058_1191820)
* Maybe, but see the suggestion in the (newly edited) answer. – [AlwaysLearning](https://superuser.com/users/146419/alwayslearning) [Mar 25 '17 at 17:03](https://superuser.com/questions/1191805/can-git-bash-and-cygwin-shell-do-the-same-things#comment1731562_1191820)

[add a comment](https://superuser.com/questions/1191805/can-git-bash-and-cygwin-shell-do-the-same-things)

1

There is a lot of overlap between the two. But ultimately they are different implementations of bash/shall in Windows.

Cygwin has a large body of optional software you can install, including multiple different programming languages, compilers, various network tools, and so on. You can also install X and many X applications.

With git bash, you get what you get, and it's a bit more difficult to install additional tools. I actually have both on my workstation, because there's things I can't do in git bash. I've had very mixed success calling cygwin apps from the git bash window, too.

Also, cygwin is a full posix environment, not just a bash interpreter, so you can often compile non-distributed code within it, so for example if you found the source of a nice Linux tool you found, you can usually compile that tool for Cygwin. Not so much for git bash. And as I alluded above, even if you do compile it for Cygwin it likely won't work in git bash.

Other than that they are mostly very similar. It's kind of funny to see that git bash includes the cygpath command for converting Windows and Posix file paths, because git bash afaik is based on MinGW, which is a distinct solution from Cygwin.